

Spatially dispersionless, unconditionally stable FC-AD solvers for variable-coefficient PDEs

O. P. Bruno* and A. Prieto†

Abstract

We present fast, spatially dispersionless and unconditionally stable high-order solvers for Partial Differential Equations (PDEs) with *variable coefficients* in general smooth domains. Our solvers, which are based on (i) A certain “Fourier continuation” (FC) method for the resolution of the Gibbs phenomenon, together with (ii) A new, preconditioned, FC-based solver for two-point boundary value problems (BVP) for variable-coefficient Ordinary Differential Equations, and (iii) An Alternating Direction strategy, generalize significantly a class of FC-based solvers introduced recently for constant-coefficient PDEs. The present algorithms, which are applicable, with high-order accuracy, to variable-coefficient elliptic, parabolic and hyperbolic PDEs in general domains with smooth boundaries, are unconditionally stable, do not suffer from spatial numerical dispersion, and they run at FFT speeds. The accuracy, efficiency and overall capabilities of our methods are demonstrated by means of applications to challenging problems of diffusion and wave propagation in heterogeneous media.

Keywords: High-order methods, Alternating Direction Implicit schemes, numerical dispersion, variable coefficient problems.

1 Introduction

We present fast, spatially dispersionless and unconditionally stable high-order solvers for Partial Differential Equations (PDEs) with *variable coefficients* in general smooth domains. Our algorithms, which generalize significantly a class of solvers [6, 12] introduced recently for constant-coefficient PDEs, are based on (i) A certain “Fourier continuation” (FC) method [6] for the resolution of the Gibbs phenomenon, together with (ii) A new, preconditioned, FC-based solver for two-point boundary value problems (BVP) for variable-coefficient Ordinary Differential Equations (ODE), and (iii) The Alternating Direction Implicit (ADI) methodology [8, 14]. One of the main enabling elements in our overall FC-AD algorithm (Fourier-Continuation Alternating-Directions) is the new solver (ii) for two-point boundary value problems with variable coefficients. Relying on preconditioners that result from inversion of oversampled finite-difference matrices together with a new methodology for enforcement of boundary conditions and the iterative linear algebra solver GMRES, this algorithm produces rapidly the solutions required for FC-AD time-stepping in the variable-coefficient context. (A non-oversampled finite-difference preconditioner related to but different from the one used here was introduced in [15] in the context of orthogonal collocation methods for equations with constant coefficients.) The resulting PDE solvers, which in practice are found to be unconditionally stable, do not suffer from spatial numerical dispersion and they run at a computational cost that grows as $\mathcal{O}(N \log_2 N)$ with the size N of the computational grid. A variety of examples presented in this paper demonstrate the accuracy, speed and overall capabilities of the proposed methodology.

The variable-coefficient FC-AD algorithms introduced in this paper enjoy all the good qualities associated with the constant coefficient solvers presented in [6, 12]: the performance of the new solvers compare

*Applied and Computational Mathematics, California Institute of Technology, Pasadena, CA 91125, USA – bruno@acm.caltech.edu

†Departamento de Matemáticas, Univ. da Coruña, 15071 A Coruña, Spain.

favourably, in terms of accuracy and speed, with those associated with previous approaches; a detailed discussion in these regards can be found in the introductory sections of [6, 12]. In particular, in this paper we demonstrate the high-order, essentially dispersionless character of the new FC-AD solvers by means of solutions to parabolic and hyperbolic problems. For example, the results presented in Section 7.3.1, which include FC-AD fixed-accuracy solutions at fixed numbers of points-per-wavelength for problems of sizes ranging from one to one-hundred wavelengths in size, demonstrate the spatial dispersionlessness of the FC-AD algorithm for hyperbolic equations. Further, as shown in Table 1, for example, the proposed FC-AD algorithm can evolve a solution characterized by one-million spatial unknowns in a computing time of approximately 1.5 seconds per time step in a single-core run.

The remainder of this paper is organized as follows: after the introduction in Section 2.1 of the variable-coefficient PDEs we consider, Section 2.2 details the ADI approximations we employ. Section 3 presents the FC method, including, in Section 3.3, a special version of the FC algorithm we need to tackle variable-coefficient differential equations. Sections 4 and 5 then describe our new FC-based solver for two-point BVP, which include 1) Iterative FC-based solvers for periodic ordinary differential equations in a certain “continued” periodic context, and 2) Techniques that enable enforcement of boundary conditions in presence of either sharp or diffuse boundary layers. The combined FC/ADI scheme is described in Section 6. The overall properties of the resulting FC-AD PDE solver, finally, are demonstrated in Section 7 through a variety of numerical results.

2 Preliminaries

2.1 Variable coefficients PDEs

This paper presents FC-based solvers for linear equations containing time-independent but spatially variable coefficients. While the methods we present are applicable to any partial differential equation for which an ADI splitting is available, for definiteness we focus on the basic variable-coefficient parabolic and hyperbolic problems

$$\begin{cases} \alpha \partial_t u - \operatorname{div}(\beta \operatorname{grad} u) = f & \text{in } \Omega \times (0, T), \\ u = g & \text{on } \partial\Omega \times (0, T), \\ u = u_0 & \text{in } \Omega \times \{0\}, \end{cases} \quad (1)$$

and

$$\begin{cases} \alpha \partial_t^2 u - \operatorname{div}(\beta \operatorname{grad} u) = f & \text{in } \Omega \times (0, T), \\ u = g & \text{on } \partial\Omega \times (0, T), \\ u = u_0 & \text{in } \Omega \times \{0\}, \\ \partial_t u = u_1 & \text{in } \Omega \times \{0\} \end{cases} \quad (2)$$

in a bounded open set Ω with smooth boundary $\partial\Omega$ and within the time interval $(0, T)$. Here α , β , u_0 , u_1 , f and g are sufficiently regular functions defined in Ω ; additionally the coefficient functions α and β are assumed to satisfy the coercivity conditions

$$\begin{aligned} \alpha_0 \leq \alpha \leq \alpha_1 & \quad \text{in } \Omega, \\ \beta_0 \leq \beta \leq \beta_1 & \quad \text{in } \Omega \end{aligned}$$

for some positive constants α_0 , α_1 , β_0 and β_1 , while the initial and source functions are required to verify the relevant compatibility conditions in $\partial\Omega \times \{0\}$, namely, $u_0 = g$ for the parabolic problem (1) and $u_0 = g$, $u_1 = \partial_t g$ for the hyperbolic problem (2).

2.2 Alternating Direction schemes

This section presents the ADI splitting schemes we use for the solution of the PDE problems (1) and (2). These splitting schemes result as adequate generalizations of the Peaceman-Rachford scheme [14] for diffusion

problems and the non-centered scheme [12] for the wave equation to the present variable-coefficient context. In what follows $\Delta t > 0$ denotes the time step used in the computational time interval $[0, T]$; it is assumed that $n_{\max} \Delta t = T$ for a certain positive integer n_{\max} . Letting $t_n = n \Delta t$ for integer and even fractional values of n (e.g., $t_{n+\frac{1}{4}} = (n + \frac{1}{4}) \Delta t$), we have, in particular, $t_{n_{\max}} = T$.

2.2.1 Diffusion equation

Given the initial values u_0 , the right hand-sides $f^{n+\frac{1}{4}}(x, y) = f(x, y, t_{n+\frac{1}{4}})$ and $f^{n+\frac{3}{4}}(x, y) = f(x, y, t_{n+\frac{3}{4}})$, and the boundary values $g^{n+1}(x, y) = g(x, y, t_{n+1})$, the exact solution $\phi^n(x, y) = u(x, y, t_n)$ for $n = 1, \dots, n_{\max}$ of the diffusion problem (1) satisfies the Crank-Nicolson relation [6, 14]

$$\begin{cases} \alpha \frac{\phi^{n+1} - \phi^n}{\Delta t} - \operatorname{div} \left(\beta \operatorname{grad} \frac{\phi^{n+1} + \phi^n}{2} \right) = \frac{f^{n+\frac{1}{4}} + f^{n+\frac{3}{4}}}{2} + \mathcal{O}(\Delta t^2) & \text{in } \Omega, \\ \phi^0 = u_0 & \text{in } \Omega, \\ \phi^{n+1} = g^{n+1} & \text{in } \partial\Omega. \end{cases} \quad (3)$$

Following [14], an ADI scheme can be obtained from the Crank-Nicolson iteration: denoting by u^{n+1} the corresponding approximation of ϕ^{n+1} (for integer values $n = 1, 2, \dots$) and using the intermediate quantity $u^{n+\frac{1}{2}}$, the ADI scheme is embodied in the equations

$$u^{n+\frac{1}{2}} - \frac{\Delta t}{2\alpha} \partial_x (\beta \partial_x u^{n+\frac{1}{2}}) = u^n + \frac{\Delta t}{2\alpha} \partial_y (\beta \partial_y u^n) + \frac{\Delta t}{2\alpha} f^{n+\frac{1}{4}} \quad \text{in } \Omega, \quad (4)$$

(with boundary condition $u^{n+\frac{1}{2}} = g^{n+\frac{1}{2}}$ on $\partial\Omega$) and

$$u^{n+1} - \frac{\Delta t}{2\alpha} \partial_y (\beta \partial_y u^{n+1}) = u^{n+\frac{1}{2}} + \frac{\Delta t}{2\alpha} \partial_x (\beta \partial_x u^{n+\frac{1}{2}}) + \frac{\Delta t}{2\alpha} f^{n+\frac{3}{4}} \quad \text{in } \Omega, \quad (5)$$

(with boundary condition $u^{n+1} = g^{n+1}$ on $\partial\Omega$); cf. [6].

An algorithm based on the ADI iteration (4)-(5) can be conveniently obtained as a sequence of four operations involving two additional auxiliary quantities w^n and $w^{n+\frac{1}{2}}$:

(D1) Initialize u^0 and w^0 as

$$\begin{cases} u^0 = u_0 & \text{in } \Omega, \\ w^0 = \left(1 + \Delta t \frac{\partial_y \beta}{2\alpha} \partial_y + \Delta t \frac{\beta}{2\alpha} \partial_y^2 \right) u^0 & \text{in } \Omega. \end{cases}$$

Then, for $n = 0, 1, \dots, n_{\max}$,

(D2) Obtain $u^{n+\frac{1}{2}}$ by solving the boundary value problem

$$\begin{cases} \left(1 - \Delta t \frac{\partial_x \beta}{2\alpha} \partial_x - \Delta t \frac{\beta}{2\alpha} \partial_x^2 \right) u^{n+\frac{1}{2}} = w^n + \frac{\Delta t}{2\alpha} f^{n+\frac{1}{4}} & \text{in } \Omega, \\ u^{n+\frac{1}{2}} = g^{n+\frac{1}{2}} & \text{on } \Omega. \end{cases} \quad (6)$$

(D3) Update $w^{n+\frac{1}{2}}$ according to

$$w^{n+\frac{1}{2}} = 2u^{n+\frac{1}{2}} - w^n - \frac{\Delta t}{2\alpha} f^{n+\frac{1}{4}} \quad \text{in } \Omega,$$

and, finally,

(D4) Obtain u^{n+1} by solving the boundary value problem

$$\begin{cases} \left(1 - \Delta t \frac{\partial_y \beta}{2\alpha} \partial_y - \Delta t \frac{\beta}{2\alpha} \partial_y^2 \right) u^{n+1} = w^{n+\frac{1}{2}} + \frac{\Delta t}{2\alpha} f^{n+\frac{3}{4}} & \text{in } \Omega, \\ u^{n+1} = g^{n+1} & \text{on } \partial\Omega. \end{cases} \quad (7)$$

The ADI scheme thus requires solution of the one-dimensional boundary value problems (D2) and (D4) (see Section 4) and solution updates (D1) and (D3). Each one of these operations involves differential operators with respect to a single spatial variable—as it behooves an ADI discretization [13].

Remark 1. An implementation of this algorithm for the case in which the coefficients α and β are constant was put forth in [6]. In that reference it was noted that, for $n \in \mathbb{N}$, u^{n+1} is a globally second order accurate approximation of the exact solution of the diffusion equation (1): the error at any fixed time step $t = t^*$ is a quantity of order $\mathcal{O}(\Delta t)^2$. This is in spite of the approximation $u^{n+\frac{1}{2}} = g^{n+\frac{1}{2}}$, which is necessary to enable applicability to complex domains, and which induces a second-order local truncation error in step (D2); see [6, Remark 3.2] for details. Numerical experiments we present in this paper reveal once again a second-order global error in the solutions resulting from the scheme above. As shown in reference [6] and Section 7.2, further, solutions of higher order of temporal accuracy can be extracted from the solutions produced by the ADI scheme above by means of the Richardson extrapolation methodology.

2.2.2 Wave equation

To derive our Alternating Direction scheme for the linear wave problem (2), in turn, we follow [12] and note that the exact solution $\phi^n(x, y) = u(x, y, t_n)$ satisfies the discrete relation

$$\begin{cases} \alpha \frac{\phi^{n+1} - 2\phi^n + \phi^{n-1}}{\Delta t^2} - \operatorname{div}(\beta \operatorname{grad} \phi^{n+1}) = f^{n+\frac{1}{2}} + \mathcal{O}(\Delta t) & \text{in } \Omega, \\ \phi^0 = u_0 & \text{in } \Omega, \\ \phi^1 = u_0 + \Delta t u_1 + \mathcal{O}(\Delta t) & \text{in } \Omega, \\ \phi^{n+1} = g^{n+1} & \text{in } \partial\Omega, \end{cases} \quad (8)$$

where, once again, $f^{n+\frac{1}{2}}(x) = f(x, y, t_{n+\frac{1}{2}})$ and $g^{n+1}(x) = g(x, y, t_{n+1})$. Following [12] we split the stiffness term and thus obtain the ADI time-stepping scheme

(W1) Initialize u^0 and u^1 as

$$\begin{cases} u^0 = u_0 & \text{in } \Omega, \\ u^1 = u_0 + \Delta t u_1 & \text{in } \Omega. \end{cases}$$

Then, for $n = 0, 1, \dots, n_{\max}$,

(W2) Obtain $w^{n+\frac{1}{2}}$ by solving the boundary value problem,

$$\begin{cases} \left(1 - \Delta t^2 \frac{\partial_x \beta}{\alpha} \partial_x - \Delta t^2 \frac{\beta}{\alpha} \partial_x^2\right) w^{n+\frac{1}{2}} = 2u^n - u^{n-1} + \frac{\Delta t^2}{\alpha} f^{n+\frac{1}{2}} & \text{in } \Omega, \\ w^{n+\frac{1}{2}} = g^{n+1} & \text{on } \partial\Omega \end{cases} \quad (9)$$

(W3) Obtain u^{n+1} as the solution of the boundary value problem

$$\begin{cases} \left(1 - \Delta t^2 \frac{\partial_y \beta}{\alpha} \partial_y - \Delta t^2 \frac{\beta}{\alpha} \partial_y^2\right) u^{n+1} = w^{n+\frac{1}{2}} & \text{in } \Omega, \\ u^{n+1} = g^{n+1} & \text{on } \partial\Omega, \end{cases}$$

Remark 2. Note that the expressions (D2)-(D4), and (W2)-(W3) of the alternating-direction ODEs for the heat and wave equations incorporate a division by the lowest-order variable-coefficient α in equations (3) and (8). This is an essential detail of our algorithm: if such a division by α is not incorporated in the algorithm, the iterative GMRES solution of these ODEs (which is presented in Section 5.1) would require large numbers of iterations. In the divided form, in contrast, the matrix of the linear-algebra problem is close to the identity for small Δt , and small numbers of GMRES iterations suffice to yield highly accurate ODE solutions. In fact, we have found in practice that the divided ODE forms give rise to small numbers of iterations even for large values of Δt .

Remark 3. It is easy to check that the scheme (W1)-(W3) is first order consistent. We refer to [12, Section 5] for a discussion of the global order of accuracy of the algorithm; in practice, and in agreement with that reference, we find the algorithm produces solutions with global first order accuracy. As shown in reference [6] and Section 7.3.2, further, solutions of higher order of temporal accuracy can be extracted from the solutions produced by the ADI scheme above by means of the Richardson extrapolation methodology.

Problems (D2), (D4), (W2) and (W3) amount to two-point BVP of the form

$$u - pu' - qu'' = f \quad \text{in } (a, b), \quad (10)$$

$$u(a) = d_a, \quad u(b) = d_b, \quad (11)$$

where the right-hand side f and the variable coefficients p and q are bounded smooth functions defined in the interval $[a, b]$, with $q > \eta > 0$ for some constant η . More precisely, the ODE coefficients and the right-hand sides of the problems (D2) and (D4), (W2) and (W3) are given by

$$\begin{cases} p(x) = \mathcal{P}^H(x, y), \quad q(x) = \mathcal{Q}(x, y), \quad f(x) = \mathcal{F}(x, y, t) & \text{in problems (D2) and (W2) } (y, t \text{ fixed}), \text{ and} \\ p(y) = \mathcal{P}^V(x, y), \quad q(y) = \mathcal{Q}(x, y), \quad f(y) = \mathcal{F}(x, y, t) & \text{in problems (D4) and (W3) } (x, t \text{ fixed}), \end{cases} \quad (12)$$

where for (D2) and (D4)

$$\mathcal{P}^H(x, y) = \Delta t \frac{\partial_x \beta(x, y)}{2\alpha(x, y)}, \quad \mathcal{P}^V(x, y) = \Delta t \frac{\partial_y \beta(x, y)}{2\alpha(x, y)}, \quad \mathcal{Q}(x, y) = \Delta t \frac{\beta(x, y)}{2\alpha(x, y)}, \quad \mathcal{F}(x, y, t) = \Delta t \frac{f(x, y, t)}{2\alpha(x, y)}, \quad (13)$$

while for (W2) and (W3)

$$\mathcal{P}^H(x, y) = \Delta t^2 \frac{\partial_x \beta(x, y)}{\alpha(x, y)}, \quad \mathcal{P}^V(x, y) = \Delta t^2 \frac{\partial_y \beta(x, y)}{\alpha(x, y)}, \quad \mathcal{Q}(x, y) = \Delta t^2 \frac{\beta(x, y)}{\alpha(x, y)}, \quad \mathcal{F}(x, y, t) = \Delta t^2 \frac{f(x, y, t)}{\alpha(x, y)}. \quad (14)$$

To solve the two-point BVP (10)-(11) numerically we utilize a discrete method based on three main elements: 1) The Fourier Continuation method (see Section 3) to produce accurate Fourier-series approximations of the ODE source terms and variable coefficients; 2) A specialized Fourier collocation method for solution of ODE boundary value problems (see Section 4.1)—which, in view of item 1), can be applied to non-periodic boundary-value problems without the accuracy degradation associated with the Gibbs phenomenon; and 3) A new strategy for the enforcement the boundary conditions in the context arising from items 1) and 2) (see Sections 4.2 and 4.3).

3 Fourier Continuation and scaled FC(Gram)

3.1 Discrete Fourier analysis background

Let $\mathcal{C}_{\text{per}}^k(a, c)$ denote the space of k -times continuously differentiable periodic functions of period $c - a$. The discrete Fourier series of $v \in \mathcal{C}_{\text{per}}^k(a, c)$ is given by

$$\mathcal{J}v(x) = \sum_{n \in \mathcal{T}(F)} \hat{v}_n e^{i \frac{2\pi n(x-a)}{c-a}} \in \mathcal{B}(a, c).$$

Here $\mathcal{T}(F) = \{n \in \mathbb{N} : -F/2 + 1 \leq n \leq F/2\}$ for F even and $\mathcal{T}(F) = \{n \in \mathbb{N} : (F-1)/2 \leq n \leq -(F-1)/2\}$ for F odd,

$$\mathcal{B}(a, c) = \left\{ g : (a, c) \rightarrow \mathbb{R} \text{ such that } g(x) = \sum_{n \in \mathcal{T}(F)} g_n e^{i \frac{2\pi n(x-a)}{c-a}} \right\} \quad (15)$$

denotes the F -dimensional space of trigonometric polynomials, and the amplitudes \hat{v}_n are given by the trapezoidal-rule expression

$$\hat{v}_n = \frac{1}{F} \sum_{j=1}^F v(x_j) e^{-i \frac{2\pi n(x_j - a)}{c - a}}, \quad n \in \mathcal{T}(F).$$

Note that, for conciseness, the “degree” F is not explicitly displayed in the notation $\mathbf{B}(a, c)$.

We point out that, as is well-known [2, 11],

1. An element of the set $\mathbf{B}(a, c)$ is determined uniquely by its values at the equispaced grid $a = x_1 < \dots < x_F = c - h$, where

$$x_j = a + (j - 1)h, \quad j = 1, \dots, F, \quad h = (c - a)/F, \text{ and,} \quad (16)$$

2. The discrete Fourier operator \mathcal{J} is an interpolation operator from $\mathcal{C}_{\text{per}}^k(a, c)$ into $\mathbf{B}(a, c)$, that is, $\mathcal{J}v(x_j) = v(x_j)$ for all $j = 1, \dots, F$.

3.2 Fourier Continuation and FC(Gram)

The Fourier Continuation method [4, 6] is an algorithm which, acting on a set of values $f(x_1), \dots, f(x_N)$ of a function $f : [a, b] \rightarrow \mathbb{R}$ at N equidistant points $x_j \in [a, b]$, produces a trigonometric polynomial $f^c \in \mathbf{B}(a, c)$, of a given degree F and on a given interval $[a, c] \supset [a, b]$ with $b \leq c$, which approximates the function f closely on the interval $[a, b]$. Note that the endpoints a and b are not required to belong to the Fourier grid $\{x_j\}_{j=1}^N$. Clearly, for $[a, b] = [a, c]$, the trigonometric interpolation operator \mathcal{J} leads to a naive Fourier Continuation procedure which gives rise to the well known Gibbs ringing near $x = a$ and $x = b$ —unless f is a smooth periodic function of period $(b - a)$. The selection of a larger expansion interval $[a, c]$ allows for a smooth transition between the values $f(x_j)$ near $j = N$ to the values $f(x_j)$ near $j = 1$, and thus enables highly accurate Fourier approximation in the interval $[a, b]$, avoiding the undesirable Gibbs ringing effect and related accuracy deterioration.

A number of algorithms has been introduced which provide accurate Fourier continuations for smooth non-periodic functions (including, for example, the FC(SVD) method [6], which relies on Singular Value Decompositions and the related algorithms [3, 5, 7]). Use of the spectrally accurate Fourier continuations as a component of efficient PDE solvers, however, must rely on correspondingly efficient Fourier Continuation algorithms. For that purpose, an accelerated FC method, the FC(Gram) procedure, was introduced recently [4, 6, 12]. In this section, we present a brief description of FC(Gram) method; full details can be found in [6]. A new “scaled” version of the FC(Gram) approach, which is necessary for the applications presented in this paper, is introduced in Section 3.3.

For the present description of the FC(Gram) method, without loss of generality we assume $[a, b] = [0, 1]$, as in references [6, 12]. The FC(Gram) algorithm is based on the N_Δ left-most and N_Δ right-most grid values $f(x_j)$ of f within the subintervals $[0, \Delta]$ and $[1 - \Delta, 1]$, with $\Delta = hN_\Delta$. Fixed the number of extension points N_d (belonging to $[1, 1 + d]$ with $d = h(N_d - 1)$) where $f^c(x_j)$ has to be computed, the continuation problem is reduced to seek a smooth periodic function f^c in $[a, c] = [0, 1 + d]$. For adequate reference note that our quantities N , N_Δ and N_d are denoted by n , n_Δ and n_d in the contribution [6].

To compute f^c , the left-most grid values of f in $[0, \Delta]$ are mapped in $[1 + d, 1 + d + \Delta]$. Then, they are projected on certain Gram bases of orthogonal polynomials—with orthogonality dictated by the natural discrete scalar product defined by the grid points. Finally, using the orthogonal polynomial expansion, a periodic matching function f_{match} , in the interval $[1 - \Delta, 1 + 2d + \Delta]$ is computed. This function consists of a sum of contributions of each polynomial projection (up to degree m) and blends smoothly the values at the left and right subintervals, $[1 - \Delta, 1]$ and $[1 + d, 1 + d + \Delta]$. The grid values of f_{match} in $[1, 1 + d]$ give the required grid values of f^c . This procedure can be efficiently implemented by precomputing the sets of matching functions $\{f_{\text{even}}^r\}_{r=0}^m$ and $\{f_{\text{odd}}^r\}_{r=0}^m$ associated to the even and odd pairs of the polynomial basis

in the interval $[0, 1]$ and then, each FC(Gram) continuation in an arbitrary interval can be obtained by using an affine transformation (see [6, Section 2.3] for details).

It is clear that the accuracy of the Fourier continuation operator depends on the number N of Fourier grid points contained in $[0, 1]$, the maximum degree of polynomials m used in the polynomial projections, the number N_Δ of right and left grid points, and also on the number N_d of extension points outside the interval $[0, 1]$ (see item 2 in Section 7 for an indication on the actual values of these parameters used in our numerical examples).

3.3 Scaled FC(Gram) algorithm

The FC-based PDE solvers presented in this paper rely on the FC methodology as a tool for solving two-point BVPs for ordinary differential equations with variable coefficients. As discussed in Section 5.3, to obtain an accurate solution of the relevant two-point BVPs for a given PDE problem in the present variable-coefficient context, our method requires, unlike previous FC-based approaches, use of numbers N_g of extension points and associated extension intervals of length g that vary (linearly) with N (see Section 5.2). Use of such N -dependent extension intervals in the procedure described in Section 3.2 entails evaluation of correspondingly N -dependent sets of matching functions $\{f_{\text{even}}^r\}_{r=0}^m$ and $\{f_{\text{odd}}^r\}_{r=0}^m$ —a procedure that is inelegant and computationally expensive.

To overcome this difficulty, a slightly modified “scaled” version of the FC(Gram) algorithm is introduced in this section. This approach is based on use of a set of precomputed matching functions f_{match} , as described in Section 3.2, in a fixed interval $[1, 1 + d]$ and for a certain fixed value N_d . To obtain inexpensively a matching function at a new number N_g of extension points in a new interval $[1, 1 + g]$ (where it is assumed that $g/N_g = d/N_d$), the “scaled” procedure uses as a matching function a composition of the form

$$f_{\text{match}} \circ \xi, \quad (17)$$

where $\xi : [1 - \Delta, 1 + g + \Delta] \rightarrow [1 - \Delta, 1 + d + \Delta]$ is a smooth diffeomorphism. To preserve the point values of f_{match} in $[1 - \Delta, 1]$ and $[1 + d, 1 + d + \Delta]$, the scaling function ξ is assumed to be map linearly the intervals $[1 - \Delta, 1]$ and $[1 + g, 1 + g + \Delta]$ onto $[1 - \Delta, 1]$ and $[1 + d, 1 + d + \Delta]$, respectively. Throughout this paper we use the scaling function

$$\xi(s) = 1 - \Delta + (d + 2\Delta) \text{frac} \left(\frac{s + (d - g)\phi((s - 1)/g) - 1 + \Delta}{d + 2\Delta} \right) \quad s \in [1, 1 + g], \quad (18)$$

where, letting $\lfloor x \rfloor$ denote the largest integer less than or equal to x , we have set $\text{frac}(x) = x - \lfloor x \rfloor$, and the auxiliary function ϕ is given by

$$\phi(s) = \begin{cases} 1 & \text{if } s > 1, \\ \left(1 + \exp \left(\frac{1}{s} - \frac{1}{1 - s} \right) \right)^{-1} & \text{if } s \in [0, 1], \\ 0 & \text{if } s < 0. \end{cases}$$

In what follows, the scaled FC(Gram) continuation of a function f resulting from this procedure will be denoted by either of the following symbols

$$\tilde{f} = E(f) = E_g(f). \quad (19)$$

The first two of these notations can of course be used only when the value of g is either inconsequential or implicit in the context. The scaled FC(Gram) algorithm produces Fourier continuations for varying values of g and N_g at a computational cost that is essentially the same as that required by the original FC(Gram) procedure for a fixed number N_d of extension points.

Remark 4. Once the even and odd continuation of the Gram polynomials are computed and stored in memory, the FC(Gram) and scaled FC(Gram) continuations of any function only involve the computation of m inner products of size N_Δ , which requires $2m$ sums and products, and the evaluation of a trigonometric interpolant, which is performed by means of an FFT of size $N + N_g - 1$. Since m and N_Δ are independent of N , both the scaled and un-scaled versions of the FC(Gram) algorithm run at a computational cost of $\mathcal{O}((N + N_g - 1) \log(N + N_g - 1))$ operations.

4 FC BVP solver I: particular solution and boundary conditions

This section presents an FC-based method for solution of variable-coefficient BVPs of the form (10)–(11). As explained in what follows, this approach relies on the scaled FC(Gram) method to produce a periodic extension of the non-periodic problem (10)–(11) to an interval $(a, c) \supset (a, b)$, with boundary values at the endpoints of the original interval (a, b) . As detailed in Section 4.1, a particular solution for equation (10) can easily be produced on the basis of the FC(Gram) method. The enforcement of the boundary conditions, which requires some consideration, is presented in Sections 4.2 and 4.3. In the first one of these sections a direct numerical approach is presented for the evaluation of solutions of the homogeneous problem (10)–(11) ($f = 0$) with non-zero boundary values, which can be used to correct the boundary values of a particular solution. In Section 4.3 a complementary approach is introduced, which can effectively treat challenging boundary layers and stiff equations that arise as small time steps and correspondingly small coefficients $q(x)$ are used (cf. equations (6), (7) and (9)).

4.1 FC-based particular solution

To obtain a periodic embedding to the interval $(a, c) \supset (a, b)$ of the BVP (10)–(11) we utilize Fourier continuations of the functions f , p and q . To preserve the ellipticity of the problem, however, we must ensure that the extension used for the coefficient q takes on strictly positive values. To produce such strictly positive extension of a positive function q we utilize a infinitely differentiable diffeomorphism $\eta : \mathbb{R} \rightarrow (C_1, C_2)$ such as, for instance, $\eta(s) = C_1 + (C_2 - C_1)(1 + \arctan(s))/2$. Using such a diffeomorphism we define certain “limited extensions” of a positive function q by means of the expression

$$\tilde{q}^\ell = \eta \circ E(\eta^{-1} \circ q), \quad (20)$$

where E denotes the Fourier continuation procedure defined in (19). The expression (20) ensures that the values of the periodic extensions \tilde{q}^ℓ are bounded by above and below by the positive values C_1 and C_2 , respectively. Hence, the “periodic embedding” of the ODE problem (10)–(11) is given by

$$u - \tilde{p}u' - \tilde{q}^\ell u'' = \tilde{f} \quad \text{in } (a, c), \quad (21)$$

where, in accordance with equation (19), \tilde{p} and \tilde{f} denote the scaled FC(Gram) continuation of the functions p and f .

To produce approximate periodic solutions of period $c - a$ of the ODE problem (10), our algorithm relies on Fourier collocation: a numerical solution of the form

$$u_{\tilde{f}}(x) = \sum_{n \in \mathcal{T}(N+N_d-1)} \hat{u}_n e^{i \frac{2\pi n(x-a)}{c-a}} \in \mathcal{B}(a, c)$$

is sought that satisfies (10) at each one of the grid points x_j , $j = 1, \dots, N + N_d - 1$, that is

$$u_{\tilde{f}}(x_j) - \tilde{p}(x_j) \frac{du_{\tilde{f}}}{dx}(x_j) - \tilde{q}^\ell(x_j) \frac{d^2 u_{\tilde{f}}}{dx^2}(x_j) = \tilde{f}(x_j) \quad \text{for } j = 1, \dots, N + N_d - 1. \quad (22)$$

(if problem (10)–(11) has a unique solution, this system of equations is not singular, see e.g. [2].) Since the solution $u_{\tilde{f}}$ is determined uniquely by its values $u_{\tilde{f}}(x_j)$ for $j = 1, \dots, N + N_d - 1$ the \mathbb{R}^{N+N_d-1} -vector of

point values of $u_{\tilde{f}}(x_j)$, $j = 1, \dots, N + N_d - 1$ are used as the unknowns of the problem. This linear system of equations is solved by means of the iterative solver GMRES; see Section 5.1 for details.

Remark 5. Note that, since the coefficients \tilde{p} and \tilde{q}^ℓ are variable, the matrix associated to the linear system (22) is not sparse. To avoid the expense associated with a direct solution of this linear system we utilize a preconditioned iterative solver, as described in Section 5.

Clearly, the particular solution described in this section does not generally satisfy the boundary conditions imposed in (11). The necessary corrections are described in the following section.

4.2 Boundary conditions I: exterior sources

To enforce the necessary boundary conditions in (11) we consider two auxiliary boundary value problems involving ODEs of the form (10) but with certain adequately-chosen right-hand sides $g_a, g_b \in \mathbf{B}(a, c)$. The right-hand sides g_a and g_b are taken to vanish at all discretization points in the original interval (a, b) but not to vanish in (b, c) —that is to say, the selected right-hand sides correspond to sources supported in the exterior of the physical domain $[a, b]$, see e.g. Figure 2 right. Clearly, any linear combination of the form $\tilde{f} + \lambda_a g_a + \lambda_b g_b$ with $\lambda_a, \lambda_b \in \mathbb{R}$ coincides with \tilde{f} in the part of the spatial grid contained in (a, b) . Calling u_a and $u_b \in \mathbf{B}(a, c)$ the approximate FC solutions (as described in Section 4.1) corresponding to g_a and g_b , it is clear that any linear combination of the form $\lambda_a u_a + \lambda_b u_b$ satisfies the ODE (10) with null right-hand side at each one of the collocation points $x_j \in (a, b)$. It follows that, denoting by $(\lambda_{0,a}, \lambda_{0,b})$ and $(\lambda_{1,a}, \lambda_{1,b})$ the solutions of the 2×2 linear systems

$$\begin{pmatrix} u_a(a) & u_b(a) \\ u_a(b) & u_b(b) \end{pmatrix} \begin{pmatrix} \lambda_{0,a} \\ \lambda_{0,b} \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad \text{and} \quad \begin{pmatrix} u_a(a) & u_b(a) \\ u_a(b) & u_b(b) \end{pmatrix} \begin{pmatrix} \lambda_{1,a} \\ \lambda_{1,b} \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \quad (23)$$

(as shown below, the system matrix is invertible for sufficiently fine discretizations provided the functions g_a and g_b are chosen appropriately), the functions $w_a = \lambda_{0,a} u_a + \lambda_{0,b} u_b$ and $w_b = \lambda_{1,a} u_a + \lambda_{1,b} u_b$ are approximate FC solutions of (10) with null right-hand side, which satisfy the Dirichlet boundary conditions $w_a(a) = 1$, $w_a(b) = 0$, $w_b(a) = 0$ and $w_b(b) = 1$. Thus, the Fourier series

$$u_N = u_{\tilde{f}} + (d_a - u_{\tilde{f}}(a))w_a + (d_b - u_{\tilde{f}}(b))w_b \quad (24)$$

is an approximate FC solution of ODE problem (10) that satisfies exactly the prescribed Dirichlet boundary conditions (11).

The needed invertibility of the system matrix in equation (23) for sufficiently fine discretizations indeed holds provided the functions g_a and g_b satisfy the conditions

$$\int_a^b g_a(x) dx > 0, \quad \int_a^b g_b(x) dx = 0. \quad (25)$$

To establish this fact we let U_a and U_b be the *exact* $(c-a)$ -periodic solutions of equation (10) with respective right-hand sides g_a and g_b , and, using Lemma 1 presented in Appendix A, we show at first that the “exact-solution system matrix”, that is, the matrix that results as u_a and u_b are substituted by U_a and U_b in equation (23), is invertible. Indeed, if the exact-solution system matrix is singular then the U_a and U_b satisfies $U_a(a) + \mu U_b(a) = 0$ and $U_a(b) + \mu U_b(b) = 0$ for a certain $\mu \in \mathbb{R}$. The function $V = U_a + \mu U_b$ is then a solution of the two-point BVP (10)–(11) with $f = 0$, $d_a = 0$ and $d_b = 0$, and thus, it vanishes identically since, in view of Lemma 1 this problem admits a unique solution. It then follows by the smoothness and periodicity of V that $V(b) = V'(b) = 0$ and $V(c) = V'(c) = 0$. But this is a contradiction, since Lemma 1 tells us that such solutions do not exist under the hypothesis (25). Hence, the exact-solution system matrix is invertible. The invertibility for the approximate-solution system matrix (23) for sufficiently fine grids then follows from the convergence of the FC solutions u_a and u_b to U_a and U_b as the grid-size tends to zero (see Remark 6).

Remark 6. Stability and convergence proofs, which are beyond the scope of this paper, are left for future work; here we merely note that convergence of the FC solutions to exact solutions is amply demonstrated by the numerical results presented in Sections 5 and 6.

Remark 7. For the numerical experiments presented in this work we have used the auxiliary right-hand sides

$$g_a = \phi \circ \psi,$$

$$g_b = \frac{\psi g_a}{\max_{x \in (b,c)} |\psi(x) g_a(x)|},$$

where $\psi(x) = 2x - (b+c)/(c-b)$ and $\phi(x) = \exp(1 - 1/(1 - x^2))$; see e.g. Figure 2. As required by Lemma 1 in Appendix A, the functions g_a and g_b satisfy the assumption (25), and their supports, which are contained in the interval (b, c) , do not intersect the interval (a, b) .

Remark 8. Note that the solution u and its numerical approximation u_N generally possess boundary layers [1, Ch. 7] at the interval endpoints a and b in cases in which the coefficient function \tilde{q}^ℓ in equation (22) is small—which, in view of equations (6), (7) and (9), it certainly is for ODE problems arising from the FC PDE solver for small values of Δt . The presence of such boundary layers can be appreciated by consideration of equation (24): the functions w_a and w_b provide (numerical approximations of) the boundary-layer contributions.

Remark 9. The numerical boundary-layer FC solutions w_a and w_b mentioned in Remark 8 may be affected by Gibbs-like ringing errors near the endpoints unless the underlying grid adequately resolves the exact boundary layers. To ensure the detection of a non-resolved boundary layer (so that a procedure can be used to guarantee that the discretization errors are not inherited by w_a and w_b in stiff problems), the grid size h is compared to the quantities $\sqrt{\tilde{q}^\ell(a)}$ and $\sqrt{\tilde{q}^\ell(b)}$. If the grid-size h is of the order of or smaller than these quantities, the boundary layer is well-resolved by the numerical approximations w_a and w_b and no further action is needed. Otherwise, if h is larger than $\sqrt{\tilde{q}^\ell(a)}$ and $\sqrt{\tilde{q}^\ell(b)}$, then w_a and w_b do not adequately resolve the boundary layers, and an alternative treatment is necessary (Remark 16 presents details on the actual thresholds used in our numerical examples). Such an alternative method, based on use of asymptotic expansions, is presented in Section 4.3.

4.3 Boundary conditions II: High-order asymptotic-matching expansions

As noted in the introductory paragraph of Section 4, stiff ODEs and challenging boundary layers arise in the solutions of the boundary value problem (10)–(11) as small time steps are used. In this section we present an algorithm that can resolve such boundary layers, without resort to unduly fine spatial meshes, on the basis of the method of matched asymptotic expansions [1]. For our development of this asymptotic procedure, we introduce a small positive parameter ε (that is taken to equal Δt in equation (9) and $\sqrt{\Delta t/2}$ in equations (6) and (7)), we let $p = \varepsilon^2 p_0$ and $q = \varepsilon^2 q_0$, and we re-express equation (10) in the form

$$(Lu)(x) = r(x)u(x) - \varepsilon^2 \frac{d}{dx} \left(s(x) \frac{du}{dx}(x) \right) = f(x) \quad , \quad x \in (a, b), \quad (26)$$

$$u(a) = d_a \quad , \quad u(b) = d_b, \quad (27)$$

where f and the variable coefficients s and r are bounded smooth functions defined in the interval $[a, b]$ (which satisfy the conditions $s, r > \eta > 0$ for some constant η) given by

$$p_0(x) = \frac{1}{r(x)} \frac{ds}{dx}(x), \quad q_0(x) = \frac{s(x)}{r(x)}.$$

Further, using the change of variables

$$y(x) = \int_a^x \frac{1}{s(\tau)} d\tau, \quad x \in (a, b), \quad (28)$$

equation (26) can be re-cast in the simpler form

$$\hat{L}\hat{u}(y) = \hat{r}(y)\hat{u}(y) - \varepsilon^2 \frac{d^2 \hat{u}}{dy^2}(y) = 0, \quad y \in (0, \hat{b}), \quad (29)$$

where $\hat{r}(y(x)) = r(x)s(x)$, $\hat{b} = y(b)$ and $\hat{u}(y(x)) = u(x)$

In what follows we produce, on the basis of the method of matched asymptotic expansions [1], approximate solutions $w_{a,\varepsilon}$ and $w_{b,\varepsilon}$ of the homogeneous ($f = 0$) version of equation (26), satisfying $w_{a,\varepsilon}(a) = 1$, $w_{a,\varepsilon}(b) = 0$, $w_{b,\varepsilon}(a) = 0$, $w_{b,\varepsilon}(b) = 1$; like the functions w_a and w_b introduced in the previous section, $w_{a,\varepsilon}$ and $w_{b,\varepsilon}$ can be used to correct the boundary values of a periodic FC solution u of (26). Unlike the functions introduced in the previous section, however, the asymptotic-expansion solutions $w_{a,\varepsilon}$ and $w_{b,\varepsilon}$ produce accurate solutions in the small ε regime without requiring use of fine meshes, and they can be evaluated with a fixed computational cost for arbitrarily small values of ε .

Remark 10. As mentioned above in this section, the small parameter ε included in the variable coefficients p and q corresponds to the quantities Δt and $\sqrt{\Delta t/2}$ in the time discretization of the wave and heat equations, respectively. Hence, the uniform convergence of the ODE solutions as $h \rightarrow 0$ with respect to ε guarantees the spatial convergence of the FC-AD scheme independently of the value of Δt used in the time-marching schemes.

We lay down our procedure for evaluation of $w_a(x)$ (left boundary layer); the corresponding method for $w_b(x)$ (right boundary layer) is entirely analogous. Using the change of variables $y = y(x)$ we define a new unknown $\hat{w} = \hat{w}(y)$ by $\hat{w}(y(x)) = w_a(x)$; clearly $\hat{w}(y)$ satisfies equation (29) together with the boundary conditions

$$\hat{w}(0) = 1, \quad \hat{w}(\hat{b}) = 0.$$

To obtain the approximate solution $\hat{w}(y)$ for small values of ε we use outer and inner solutions in the corresponding matched asymptotic expansion [1] associated with the left boundary layer. Setting $\varepsilon = 0$ in the ODE operator \hat{L} and using the null boundary condition at $y = b$ we see that the lowest order term in the outer asymptotic expansion vanishes. In fact, as it follows from the discussion below, the outer solution vanishes to all orders:

$$\hat{w}_{\text{out}}(y) = 0, \quad y \in (0, \hat{b}).$$

In the inner region, in turn, we use the scaled variable $Y = y/\varepsilon$ and we express the inner solution $\hat{w}_{\text{in}}(Y) \sim \hat{w}(\varepsilon Y)$ and the ODE coefficient \hat{r} as asymptotic expansions

$$\hat{w}_{\text{in}}(Y) = \sum_{j=0}^k \varepsilon^j \hat{w}_j(Y), \quad \hat{r}(\varepsilon Y) = \sum_{j=0}^k \varepsilon^j \hat{r}_j Y^j. \quad (30)$$

In particular, the Taylor expansion of the ODE coefficient $\hat{r}(\varepsilon Y)$ induces a corresponding expansion for the ODE operator \hat{L} , namely $\hat{L} = \sum_{j=0}^k \varepsilon^j \hat{L}_j$, where

$$\hat{L}_0 = -\frac{d^2}{dY^2} + \hat{r}_0, \quad \hat{L}_j = \hat{r}_j Y^j, \quad j = 1, \dots, k.$$

It follows that the functions \hat{w}_j are solutions of the following ODE problems:

$$\begin{cases} \hat{L}_0 \hat{w}_0(Y) = 0, & Y \in (0, \infty), \\ \hat{w}_0(0) = 1, \\ \lim_{Y \rightarrow \infty} \hat{w}_0(Y) = 0 \end{cases} \quad (31)$$

for $j = 0$, and

$$\begin{cases} \hat{L}_j \hat{w}_j(Y) = \sum_{m=1}^j \hat{L}_m \hat{w}_{j-m}(Y) = \sum_{m=1}^j \hat{r}_m Y^m \hat{w}_{j-m}(Y), & Y \in (0, \infty), \\ \hat{w}_j(0) = 0, \\ \lim_{Y \rightarrow \infty} \hat{w}_j(Y) = 0 \end{cases} \quad (32)$$

for $j = 1, \dots, k$. It is easy to check that $\hat{w}_0(Y) = \exp(-\sqrt{\hat{r}_0}Y)$, that, similarly, \hat{w}_j can be expressed in closed form in terms of adequate combinations of exponentials and polynomials (see Algorithm 1), and finally, that, as claimed above, all terms in the outer asymptotic expansion vanish.

In sum, we have

$$w_{a,\varepsilon}(x) = \sum_{j=0}^k \varepsilon^j \hat{w}_j\left(\frac{y(x)}{\varepsilon}\right), \quad (33)$$

where the functions \hat{w}_j satisfy (31) and (32); a corresponding asymptotic boundary correction function $w_{b,\varepsilon}$ can be obtained similarly. The maximum errors resulting from these approximations satisfy $\|w_a - w_{a,\varepsilon}\| = \mathcal{O}(\varepsilon^{k+1})$ and $\|w_b - w_{b,\varepsilon}\| = \mathcal{O}(\varepsilon^{k+1})$ for small ε . Using these asymptotic boundary correction functions we then obtain an approximate FC solution of ODE problem (10), which satisfies exactly the prescribed Dirichlet boundary conditions (11), by means of the expression

$$u_N = u_{\tilde{f}} + (d_a - u_{\tilde{f}}(a))w_{a,\varepsilon} + (d_b - u_{\tilde{f}}(b))w_{b,\varepsilon}. \quad (34)$$

Remark 11. Except for the accuracy tests presented in Section 5.3, all numerical examples presented in this paper use asymptotic expansions of order $k = 3$ for evaluation of $w_{a,\varepsilon}$ and $w_{b,\varepsilon}$ whenever asymptotic expansions are needed for evaluation of boundary corrections. The selection of the value $k = 3$ is based, precisely, on a series of numerical tests presented in Section 5.3.

We conclude this section with a description of the algorithm we use for evaluation of the terms in the asymptotic series (33). At first a Taylor series centered at $y = 0$ for the ODE coefficient \hat{r} is obtained, either from a closed form expression or by numerical differentiation of an associated Fourier continuation series: the derivatives needed to evaluate

$$\hat{r}_j = \frac{d^j \hat{r}}{dy^j}(0) = \left(s(x) \frac{d}{dx} \right)^j (r(x)s(x)) \Big|_{x=a} \quad \text{for } j = 0, \dots, k. \quad (35)$$

can be accurately produced by differentiation of limited extensions provided by the scaled FC(Gram) procedure, that is, by replacing r and s by \tilde{r}^ℓ and \tilde{s}^ℓ , respectively (see Subsection 4.1). Hence, only the grid values of the ODE coefficients are necessary to approximate the Taylor coefficients. This numerical differentiation procedure was used for all of the examples requiring use of asymptotic expansions in Sections 5.3 and 7.

Additionally, it is also necessary to provide an accurate numerical approximation of the primitive (28) that defines the change of variables $y(x)$. Our algorithm produces this primitive by relying on the Fourier continuation \tilde{s}^ℓ . Using this function we obtain the Fourier series

$$\frac{1}{\tilde{s}^\ell}(x) = \sum_{n \in \mathcal{T}(N+N_d-1)} \sigma_n e^{i \frac{2\pi n(x-a)}{c-a}} \in \mathcal{B}(a, c)$$

(by means of a Fast Fourier transform (FFT)), from which integration is straightforward:

$$y_N(x) = \int_a^x \left(\frac{1}{\tilde{s}^\ell(\tau)} - \sigma_0 \right) d\tau + (x-a)\sigma_0, \quad x \in (a, b). \quad (36)$$

In summary, the pseudocode in Algorithm 1 outlines the procedure we use for evaluation of the grid values of the asymptotic expansion $w_{a,\varepsilon}$ of order k , from two given vectors of grid values of the variable coefficients r and s .

Algorithm 1 Evaluation of the k -th order asymptotic expansion $w_{a,\varepsilon}$ for the ODE problem (26)–(27) with variable coefficients r and s .

Given the grid values of the functions r, s in (a, b)

$\hat{r}_0, \dots, \hat{r}_k \leftarrow \text{taylor_coef}(k, r, s)$ // compute the Taylor coefficients in equation (35)
 $\hat{w}_0, \dots, \hat{w}_k \leftarrow \text{ode_solve}(\hat{r}_0, \dots, \hat{r}_k)$ // compute analytically the ODE solutions (31) and (32)

$y_N \leftarrow \text{int_fft}(1/\tilde{s}^\ell)$ // integrate $1/\tilde{s}^\ell$ using a zero-mean Fourier series (36)
 $\hat{w}_{\text{in}} \leftarrow \text{expand}(\hat{w}_0, \dots, \hat{w}_k)$ // evaluate the sums (30)
 $w_{a,\varepsilon} \leftarrow \text{compose}(\hat{w}_{\text{in}}, y_N/\varepsilon)$ // compose the inner expansion with $Y = y_N/\varepsilon$

Return the grid values of $w_{a,\varepsilon}$

4.4 Discrete ODE operators

Here we introduce two operators \tilde{A}_N and \tilde{B}_N associated with the ODE solvers described in the previous sections; use of these operators facilitates the introduction of the overall PDE solvers in Section 6. Briefly, the first one of these is the “ODE solution operator” (described in Sections 4.1 through 4.3) which, given coefficients, right-hand sides and boundary values, produces approximate grid values $\mathbf{u} = (u_N(x_1), \dots, u_N(x_N))^t$ of the solution of the two-point BVP (10)–(11). The operator \tilde{B}_N , amounts, simply, to evaluation of the ODE differential operator, for given ODE coefficients, for a given discrete function \mathbf{w} .

More precisely, given the equispaced grid points x_1, \dots, x_N in the domain (a, b) of the two-point boundary value problem (10)–(11) and the grid values $\mathbf{p} = (p_1, \dots, p_N)^t \in \mathbb{R}^N$ and $\mathbf{q} = (q_1, \dots, q_N)^t \in \mathbb{R}^N$, of the ODE variable coefficients p and q , the linear operator (matrix) $\tilde{A}_N = \tilde{A}_N(\mathbf{p}, \mathbf{q})$ maps the grid values $\mathbf{f} = (f_1, \dots, f_N)^t \in \mathbb{R}^N$ of the right-hand side f and the Dirichlet data $\mathbf{d} = (d_a, d_b)^t \in \mathbb{R}^2$ into the approximate point values $\mathbf{u} = (u_N(x_1), \dots, u_N(x_N))^t$ of the ODE solution as defined in Sections 4.1 through 4.3:

$$(\mathbf{f}, \mathbf{p}, \mathbf{q}, \mathbf{d}) \in \mathbb{R}^{3N+2} \mapsto \mathbf{u} = \tilde{A}_N(\mathbf{p}, \mathbf{q}) \begin{pmatrix} \mathbf{f} \\ \mathbf{d} \end{pmatrix} \in \mathbb{R}^N. \quad (37)$$

Analogously, the matrix $\tilde{B}_N = \tilde{B}_N(\mathbf{p}, \mathbf{q})$ maps the grid values $\mathbf{w} = (w_1, \dots, w_N)^t \in \mathbb{R}^N$ into the grid point values $\mathbf{g} = (g_1, \dots, g_N)^t$ resulting from evaluation of the periodic ODE operator on the left-hand side of (21) on \mathbf{w} , that is to say,

$$(\mathbf{w}, \mathbf{p}, \mathbf{q}) \in \mathbb{R}^{3N} \mapsto \mathbf{g} = \tilde{B}_N(\mathbf{p}, \mathbf{q}) \mathbf{w} \in \mathbb{R}^N, \quad (38)$$

where, letting $w \in \mathcal{B}(a, c)$ denote the scaled FC(Gram) continuation obtained from the grid values \mathbf{w} , we have set

$$g_j = w(x_j) - \tilde{p}(x_j) \frac{dw}{dx}(x_j) - \tilde{q}^\ell(x_j) \frac{d^2w}{dx^2}(x_j) \quad \text{for } j = 1, \dots, N.$$

Remark 12. Note that, 1) As pointed out in Section 3.2, the endpoints a and b are not required to belong to the Fourier-continuation grid $\{x_j\}_{j=1}^N$ (and, in fact, they are chosen not to belong to the spatial grid x_1, \dots, x_N used in the context of the ODE operators considered in this section), and, 2) The ODE evaluation inherent in the operator \tilde{B}_N does not involve the Dirichlet boundary data d_a and d_b , since neither a nor b belong to the spatial grid used.

Remark 13. Note that the operators \tilde{A}_N and \tilde{B}_N depend on the interval endpoints a and b , although the notations (37)–(38) do not make this dependence explicit. This fact is relevant in the context of Section 6, where the endpoints a and b generally change as varying Cartesian lines and corresponding ODE domains are used as part of an alternating-direction PDE solution procedure.

4.5 Boundary projections

In order to ensure stability and convergence, the PDE solvers described in Section 6 utilize slightly modified versions of the ODE operators \tilde{A}_N and \tilde{B}_N introduced in the previous section; the resulting modified operators, which incorporate prescriptions put forth in [6] and [12], are denoted by A_N and B_N . Here we sketch

the projection and correction procedures that produce A_N and B_N from \tilde{A}_N and \tilde{B}_N ; full details in these regards can be found in reference [6].

Letting \mathbf{v} denote $\mathbf{v} = \tilde{A}_N(\mathbf{p}, \mathbf{q})(\mathbf{f}, \mathbf{d})^t$ (resp. $\mathbf{v} = \tilde{B}_N(\mathbf{p}, \mathbf{q})\mathbf{w}$), the vector $A_N(\mathbf{p}, \mathbf{q})(\mathbf{f}, \mathbf{d})^t$ (resp. $B_N(\mathbf{p}, \mathbf{q})\mathbf{w}$) is defined as the result of an application to the vector \mathbf{v} of boundary projections and corrections resulting from the following procedure:

- (a) With reference to Section 3.2, construct a vector \mathbf{v}^p of length N by replacing the N_Δ left-most and N_Δ right-most values of $\mathbf{v} = (v_1, \dots, v_N)^t$ by the corresponding values of certain polynomial approximants, the “open Gram polynomial projections”, that result from application of the FC(Gram) procedure to the corresponding values $(v_1, \dots, v_{N_\Delta})^t$ and $(v_{N_\Delta-N+1}, \dots, v_N)^t$, respectively. These corrections are called “open” as they do not involve the two endpoints of the interval $[a, b]$; see [6].
- (b) Analogously, construct a vector \mathbf{v}^b of length N by replacing the N_Δ left-most and N_Δ right-most values of $\mathbf{v} = (v_1, \dots, v_N)^t$ by the corresponding values of certain polynomial approximants, the “closed Gram polynomial projections”, that result from application of the FC(Gram) procedure to the $N_\Delta + 1$ -tuples $(d_a, v_1, \dots, v_{N_\Delta})^t$ and $(v_{N_\Delta-N+1}, \dots, v_N, d_b)^t$, respectively, where d_a and d_b are user-provided values (which, in the context of the PDE solver of Section 6, are given by the PDE boundary values on the corresponding horizontal or vertical ADI lines). These corrections are called “closed” as they involve the two endpoints of the interval $[a, b]$; see [6].
- (c) Using a second-order centered finite difference scheme obtain a discrete solution $\boldsymbol{\nu} = (\nu_1, \dots, \nu_N)^t$ of the two-point BVP (10)-(11) with right-hand side $\mathbf{f} - E(\mathbf{f})$ (see equation 19) and construct the open projection $\boldsymbol{\nu}^p$ of $\boldsymbol{\nu}$, using the prescription in item (a) above.
- (d) The vector $A_N(\mathbf{p}, \mathbf{q})(\mathbf{f}, \mathbf{d})^t$ (resp. $B_N(\mathbf{p}, \mathbf{q})\mathbf{w}$) is now produced as the linear combination $(1 - \chi)\mathbf{v}^p + \chi\mathbf{v}^b + \boldsymbol{\nu} - \boldsymbol{\nu}^p$, where

$$\chi = \min \left\{ 1, \frac{N_d - 2}{h^2} \max_{x \in (a, c)} \tilde{q}^\ell(x) \right\}. \quad (39)$$

(The definition (39) is a direct generalization of a formula put forth in [6] for the corresponding constant coefficient two-point BVP.)

5 FC BVP solver II: implementation and numerical results

This section consists of four subsections, Sections 5.1 through 5.4, which discuss, in turn, 1) A method of solution of the linear systems introduced in Sections 4.1 and 4.2 via preconditioned GMRES; 2) Selection of parameters needed for the scaled FC(Gram) algorithm described in Section 3.3; 3) The accuracy inherent in the resulting scaled FC BVP solver; and, 4) The computational cost of the ODE solution algorithm with parameters as in point 3).

5.1 Periodic ODE solution via preconditioned GMRES

The discretization (22) of the periodic variable coefficient ODE (21) gives rise to the *full* linear system of equations

$$\tilde{B}_N(\mathbf{p}, \mathbf{q})\mathbf{w} = \mathbf{g} \quad (40)$$

associated with the matrix $\tilde{B}_N(\mathbf{p}, \mathbf{q})$ defined in equation (38). The solution of such linear systems can be obtained efficiently by relying on the iterative linear algebra solver GMRES [16], on the basis of 1) FFT-based evaluation of forward maps given by the matrix $\tilde{B}_N(\mathbf{p}, \mathbf{q})$, and 2) Use of a finite-difference based preconditioner—as described in what follows.

The pseudocode in Algorithm 2 outlines our algorithm for evaluation of the operator $\tilde{B}_N(\mathbf{p}, \mathbf{q})\mathbf{w}$ for a given vector \mathbf{w} of grid values. Assuming that the point values \mathbf{p} and \mathbf{q} of the variable coefficients have been obtained a priori, the ODE evaluation involves an application of the scaled FC(Gram) algorithm to

\mathbf{w} followed by computation of the point values of its first and second derivatives and multiplication by the variable coefficients. In order to mitigate aliasing effects in evaluation of products, oversampling by a factor of two (obtained from an adequately zero-padded FFTs) is used in the precomputed coefficients \mathbf{p} and \mathbf{q} as well as the function values and derivatives arising from the vector \mathbf{w} .

Algorithm 2 Evaluation of the function $\mathbf{g} = \tilde{B}_N(\mathbf{p}, \mathbf{q})\mathbf{w}$: application of the ODE operator for a vector \mathbf{w} of function grid values

Given the function grid values \mathbf{w}

```

 $\tilde{\mathbf{w}} \leftarrow \text{scaled\_fcgram}(\mathbf{w})$            // apply the scaled FC(Gram) procedure
 $\hat{\mathbf{w}} \leftarrow \text{fft}(\tilde{\mathbf{w}})$                // apply the Fourier transform
 $\hat{\mathbf{g}} \leftarrow \text{oversample}(\hat{\mathbf{w}})$          // oversample the Fourier series (by zero-padding) by a factor of two
 $(\hat{\mathbf{g}}', \hat{\mathbf{g}}'') \leftarrow \text{fourier\_diff}(\hat{\mathbf{g}})$  // compute Fourier space derivatives
 $(\mathbf{g}', \mathbf{g}'') \leftarrow \text{ifft}(\hat{\mathbf{g}}', \hat{\mathbf{g}}'')$  // evaluate derivatives in physical space
 $\mathbf{g} \leftarrow \mathbf{g} - \mathbf{p}\mathbf{g}' - \mathbf{q}\mathbf{g}''$        // evaluate the ODE operator

```

Return the grid values of \mathbf{g}

As mentioned above, our algorithm obtains the solutions of the linear systems (40) by means of a *preconditioned* GMRES solver. (A direct application of the un-preconditioned GMRES algorithm to these linear systems requires extremely large numbers of iterations to meet even modest accuracy tolerances: in the case of the two-point BVP considered in Figure 1, for example, the numbers of iterations required to obtain a residual of 10^{-10} for $N = 100, 1000$ and 3000 are 145, 1279 and 3799, respectively.) The preconditioners used, which are mentioned in point 2) above, are given by inverses of the matrices corresponding to (possibly oversampled) second-order finite difference (FD) approximations of the periodic ODE problem (21) (cf. reference [15] and associated comments in Section 1). Since the ODE variable coefficients and the right-hand side in (10) are accurately represented by their scaled FC(Gram) continuations, a zero-padding procedure can be used to evaluate the ODE coefficients and right-hand side on a spatial grid that is finer, by a certain factor N_{over} , than the one used in the original Fourier collocation discretization (22), thus easily leading to an oversampled preconditioner. With limited impact on the computing cost of the algorithm (see Section 5.4), a fine grid FD discretization can improve significantly, by a factor of $1/N_{\text{over}}^2$, the accuracy of the preconditioner—and thus, as demonstrated in Figure 1, its preconditioning capability.

Figure 1 displays the number of iterations required by the GMRES-based preconditioned iterative ODE solver described above in this section to reach a residual tolerance $\text{tol}_{\text{GMRES}}$, for various values of the preconditioner oversampling parameter N_{over} . For these examples a two-point boundary value problem (10)–(11) was considered in the interval $(a, b) = (0, 1)$ with coefficients given by $p(x) = 24x/(1 + 4x^2)$ and $q(x) = (1 + 8x^3)/(1 + 4x^2)$, and with right-hand side and the Dirichlet boundary values such that the function $u(x) = \cos(x^2 + 2)$ is an exact solution, and the FC parameters $N_d = \lfloor 26(1 + (N - 21)/100) \rfloor$ (see Section 5.2), $N_\Delta = 10$ and $m = 5$ were used.

These images demonstrate the GMRES convergence character of the preconditioned FC-based periodic ODE solver: the iteration numbers remain bounded independently of the grid size, and they decrease as more accurate preconditioners are used. For comparison purposes, results obtained through preconditioning by means of the constant coefficient ODE solver [6] (with constant coefficients equal to the average of the given variable coefficients) are shown, the corresponding results are marked as “Const. coef.” in Figure 1. Clearly, in both plots, the constant-coefficient preconditioner is less efficient than the oversampled second-order FD preconditioners.

5.2 Parameter selection for the scaled FC(Gram) algorithm of Section 3.3

The exterior-source procedure for enforcement of boundary conditions introduced in Section 4.2, requires evaluation of two right-hand sides, $g_a, g_b \in \mathcal{B}(a, c)$, with support outside the interval (a, b) (see Section 4.2). Plots in Figure 2 show our selections for the functions g_a and g_b with $N_d = 26$ (left) and $N_d = 260$ (right),

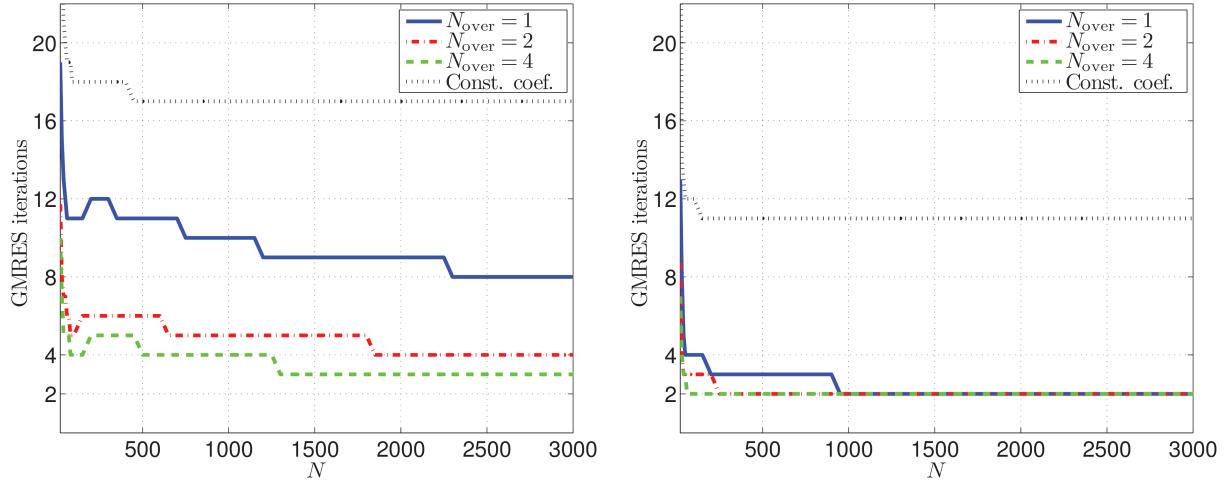


Figure 1: Number of GMRES iterations required for the solution of the linear system (22) as a function of the number N of grid points in the one-dimensional mesh, using the GMRES residual tolerances $tol_{\text{GMRES}} = 10^{-15}$ (left) and 10^{-10} (right), for several finite-difference preconditioning oversampling numbers N_{over} and a for a simple preconditioner based on a constant-coefficient approximation.

taking in both cases an $N = 1000$ point discretization of the interval (a, b) . As demonstrated on the left

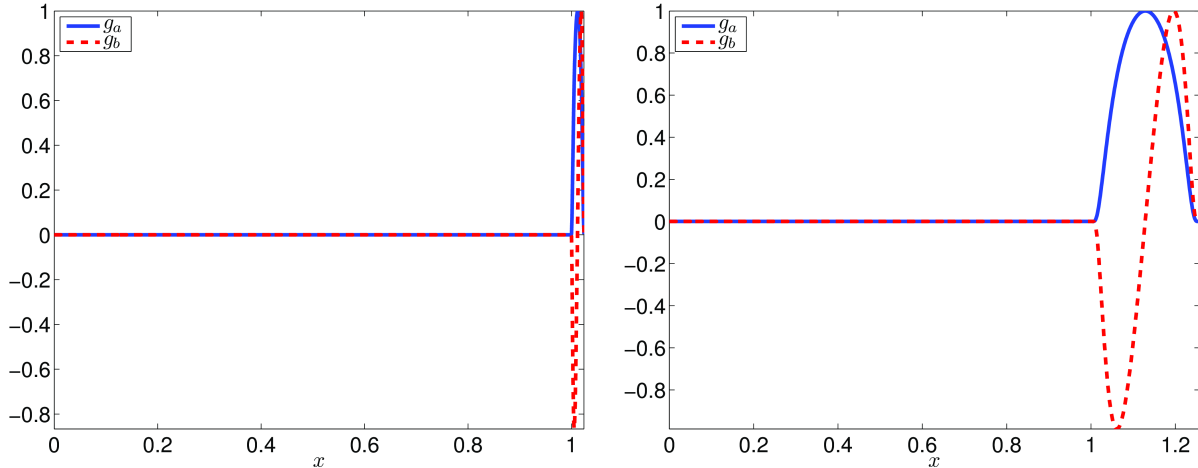


Figure 2: Auxiliary right-hand sides g_a , g_b used in the exterior-source procedure for $N_d = 26$ (left) and $N_d = 260$ (right).

portion of Figure 2, if the number N_d of grid points outside the interval (a, b) is fixed, then the size of the support of the functions g_a and g_b tends to zero and, thus, although the maximum norm of these functions equal one for all values of N_d , both functions tend to zero in the root-mean-square norm as $N \rightarrow \infty$. Thus, for fixed N_d , the solutions u_a and u_b introduced in Section 4.2, and thus the matrices of the systems (23), tend to zero as $N \rightarrow \infty$. Clearly, then, the exterior-source method put forth in Section 4.2 requires that $N_d \rightarrow \infty$ at least linearly with N as $N \rightarrow \infty$.

The scaled FC(Gram) method introduced in Section 3.3 was designed precisely to allow for evaluation

of Fourier continuation functions within a context that includes use of variable values of the parameter N_d , while still allowing for re-use of the basic matching functions $\{f_{\text{even}}^r\}_{r=0}^m$ and $\{f_{\text{odd}}^r\}_{r=0}^m$ defined in Section 3.2. The upper and lower left portions of Figure 3 display the un-scaled and scaled continuations of the function $f(x) = x^2 \cos(x^2)$ using

$$N_d = \lfloor 26(1 + (N - 21)/100) \rfloor, \quad (41)$$

where $\lfloor s \rfloor$ denote the largest integer less than or equal to s . The right portion of Figure 3 demonstrates the fact that, provided values of N_d proportional to N are used (as is done here, according to equation (41)), the relative error of the scaled FC continuations (see Remark 14) decays at the same rate as its non-scaled counterpart as N tend to infinity. Although the scaled version is somewhat less accurate than the original FC(Gram) continuations proposed in [6], the scaling to the interval $[1, 1 + g]$ implicit in equation (41) (see Section 3.3) can be implemented through use of a small set of precomputed scaled matching functions.

Remark 14. Throughout the present Section 5, the relative error of scaled and unscaled FC(Gram) continuations of given functions has been evaluated as the difference between the corresponding FC approximation and the corresponding exact function values over an equispaced grid ten times finer than the grid used in the FC procedure. (We have checked that, for the functions under consideration, use of higher grid refinement rates leads to essentially unchanged error estimates.) Clearly it is necessary to use such oversampling, as, although not exactly interpolatory, the FC procedure does arise from a least-squares approximation of function values (at N_Δ collocation points next to each one of the interval endpoints), and, thus, evaluation of errors solely at collocation points tend to produce error under-estimates. It is reasonable to expect (and we have verified in practice) that this difficulty does not arise when FC solutions of ODEs are concerned, since the FC BVP solution procedure does not seek to minimize error in function values at the collocation points. In spite of this fact, and for consistency, the solution errors presented in Figures 4 and 5 were evaluated on a grid resulting from ten-fold refinement.

Remark 15. The right-hand image in Figure 3 demonstrates that the order of convergence of the scaled FC(Gram) method is consistent with the order obtained by the original un-scaled procedure, in this case

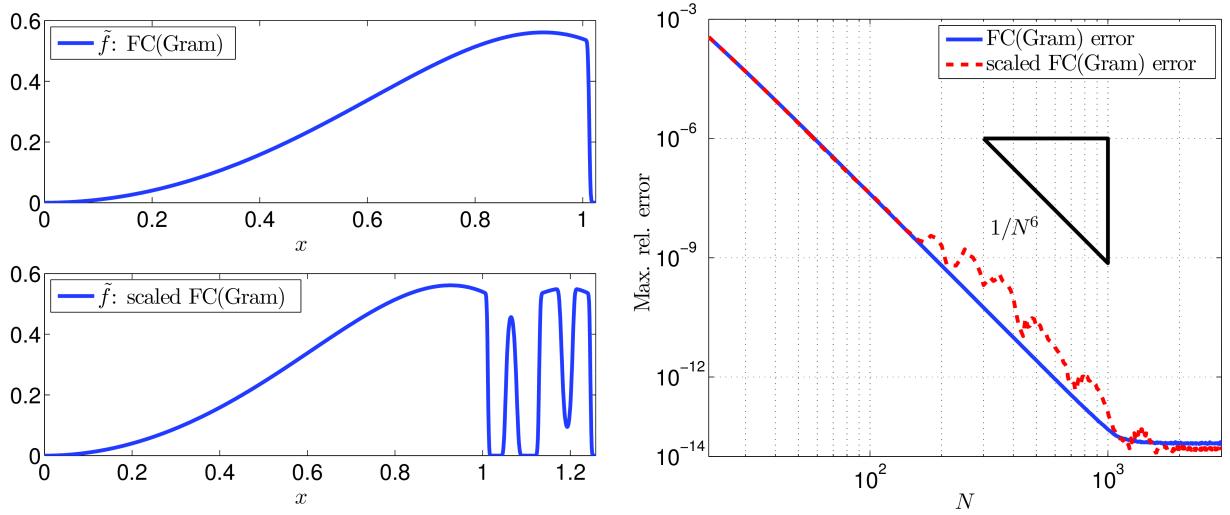


Figure 3: Fourier continuation of the function $f(x) = x^2 \cos(x^2)$ in the interval $(0,1)$ as produced by the original FC(Gram) method with $N_d = 26$ (upper-left), its scaled version using $N_d = 260$ (lower-left), and the corresponding maximum relative errors (relative to the maximum value of the original function) as a function of N (right). As demonstrated in Figure 4, the scaled algorithm is an essential element of the two-point boundary value solver introduced in Section 4.

$1/N^6$, and that both algorithms reach round-off machine error levels for approximately the same values of N . However, it is clear from this figure that the convergence of the scaled FC(Gram) algorithm is somewhat more irregular than that of the unscaled procedure. The fluctuating behavior observed in Figure 3 and accompanying accuracy loss (that amounts to as much as one digit for some values of N in this example, but does not otherwise detract from the overall convergence rate), can be traced to the fact that the scaled algorithm incorporates the composition (17)–(18) and, thus, associated frequency content, as well as the discontinuous selection (41) of the number N_d of continuation points as a function of N .

5.3 Accuracy of FC BVP solver: scaling and asymptotics

5.3.1 Scaling and convergence of the exterior-source FC BVP solver

This section demonstrates the effectiveness of the scaling procedure described in Section 3.3 with parameters as indicated in Sections 5.1 and 5.2; for the purposes of this demonstration the two-point boundary value problem introduced in Section 5.1 is considered. The left portion of Figure 4 displays in a solid blue line the maximum ODE solution error (relative to the maximum value of the ODE solution) based on use of the unscaled FC(Gram) algorithm with a continuation region containing a fixed number $N_d = 26$ of discretization points. For comparison purposes, the left-hand figure also displays the corresponding maximum relative error arising in the Fourier continuation of the ODE right-hand side using the same continuation region. Clearly, while use of a continuation region that does not grow with N does not affect the accuracy of the the FC(Gram) procedure, it does affects greatly the accuracy of the ODE solver: convergence beyond rather small values of N is not observed for the un-scaled ODE solution. However, as demonstrated in the right portion of Figure 4, use of the scaled FC(Gram) method (with, e.g., N_d given by equation (41)), restores convergence to machine precision. Notice that the error in the ODE solver is a quantity of order $(1/N)^{m+1}$, where m is the maximum polynomial degree in the Gram polynomial basis used for the scaled FC(Gram) procedure.

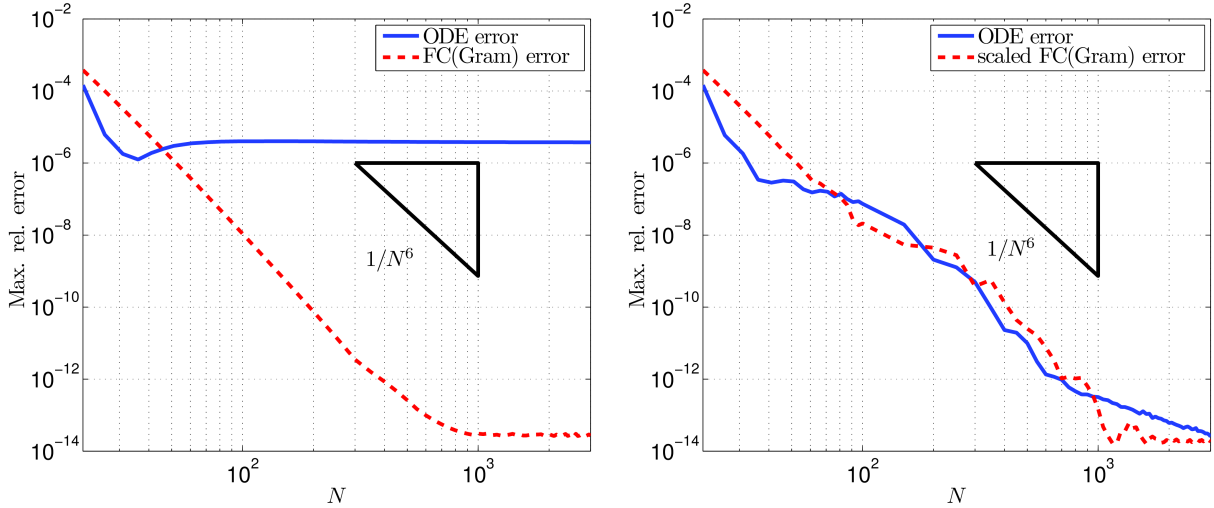


Figure 4: Maximum error (relative to the solution maximum) in the FC(Gram) approximation of the ODE right-hand side (denoted by “FC(Gram) error” in the figure) and corresponding FC ODE solution error for the two-point boundary value problem introduced in Section 5.1. Left: un-scaled procedure. Right: scaled version.

Figure 5 demonstrates the dependence of the numerical accuracy on the preconditioner used. For $N \leq 700$ all the preconditioners considered give rise to similar performance, but larger value of the oversampling

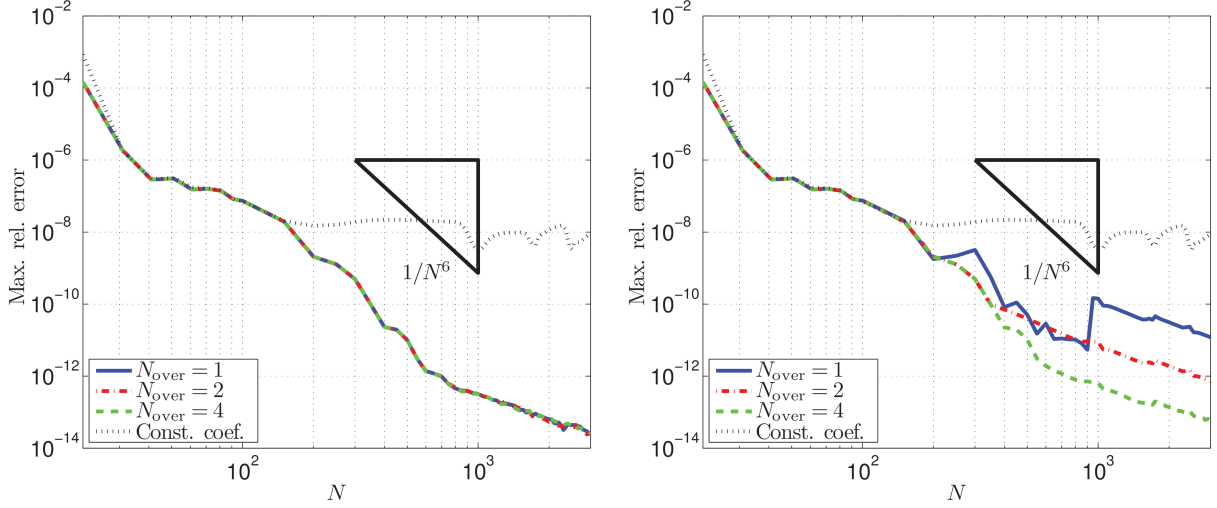


Figure 5: Maximum error (relative to the solution maximum) in the FC solution of the two-point boundary value problem considered in Section 5.3.1, as a function of the number of discretization points used, for two values of the GMRES residual tolerance: $tol_{\text{GMRES}} = 10^{-15}$ (left) and 10^{-10} (right).

parameter N_{over} do give rise to improved accuracies for the 10^{-10} GMRES tolerance and for $N \geq 1000$.

5.3.2 Convergence for stiff problems: exterior-sources and asymptotic-matching

This section demonstrates that an adequate combination of the scaling and asymptotic methods described in Sections 4.2 and 4.3 gives rise to a robust and accurate solver for *arbitrarily stiff* two-point boundary value problems under consideration (see Remark 10). For this demonstration we consider the variable-coefficient two-point BVP (10)–(11) in the interval $(0, 1)$ with $f = 0$ and with

$$p(x) = \varepsilon^2 \frac{2(1+2x)}{0.5 \log(1+2x) + 1} \quad \text{and} \quad q(x) = \varepsilon^2 \frac{(1+2x)^2}{0.5 \log(1+2x) + 1}.$$

The exact solution of this problems is given by

$$u(x) = C_1(\varepsilon) \text{Ai} \left(\varepsilon^{-\frac{2}{3}} \left(\frac{1}{2} \log(2x+1) + 1 \right) \right) + C_2(\varepsilon) \text{Bi} \left(\varepsilon^{-\frac{2}{3}} \left(\frac{1}{2} \log(2x+1) + 1 \right) \right),$$

where Ai and Bi are the Airy functions of first and second kind, respectively, and where $C_1(\varepsilon)$ and $C_2(\varepsilon)$ are chosen in such a way that $u(0) = 1$ and $u(1) = 0$. The solution u has a boundary layer at the endpoint $x = 0$ for small values of ε .

The left portion of Figure 6 displays the boundary layer solution $u(x)$ for three values of ε . The right image in the same figure presents the maximum relative errors that result from use of the exterior-source procedure (Section 4.2) and asymptotic-matching expansions (Section 4.3) of orders $k = 1, 2$ and 3 . As expected, the exterior-source and asymptotic-matching procedures are both accurate within their intended realms of applicability (ε bounded away from zero and $\varepsilon \rightarrow 0$, respectively), but they both break down otherwise. Hence, our algorithmic strategy, which relies on one procedure or the other, depending on the value of ε , results in accurate approximations for all values of $\varepsilon > 0$.

Remark 16. The adequate detection of the boundary layer in terms of the small parameter ε , and, thus, the evaluation of the threshold value that defines the limit between the (h, ε) regions for which the exterior-source and asymptotic-matching procedures are applied depends on each particular BVP. As a rule of thumb the

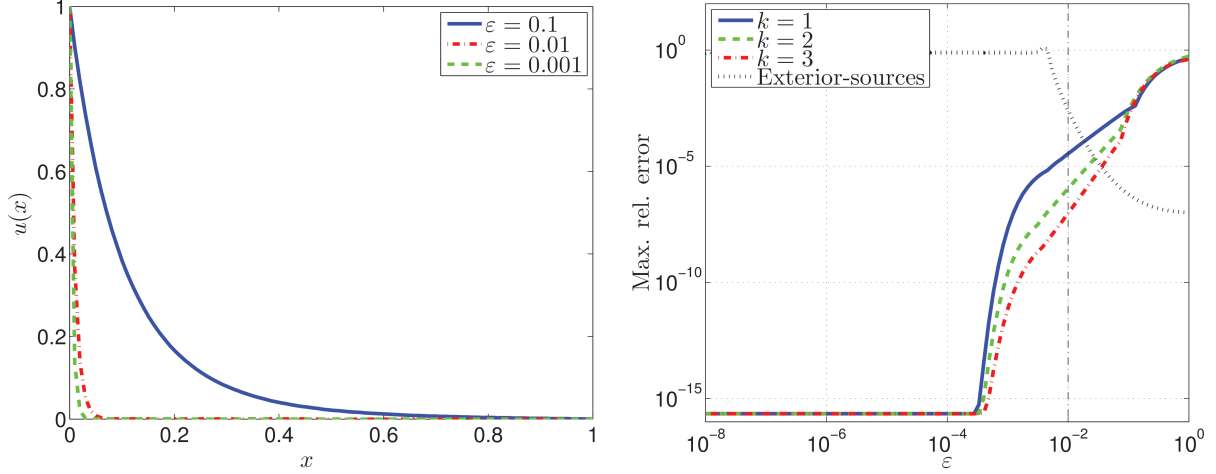


Figure 6: Boundary layer solution with small parameters $\varepsilon = 0.1, 0.01, 0.001$ (left), and maximum relative errors, as a function of ε , resulting from use of the asymptotic-matching expansions of order $k = 1, 2, 3$ as well as the exterior-source method with $h = 10^{-2}$ (right). The dash-dot vertical line on the right plot is located at $\varepsilon = h = 10^{-2}$.

threshold limit can be fixed to $\varepsilon = h$ (see Figure 6). This selection (with $\varepsilon = \sqrt{\Delta t/2}$ and $\varepsilon = \Delta t$ for the heat and wave equations, respectively) was used in all of the numerical experiments presented in this paper. Section 7.2 and, in particular, Table 1, provide an indication of the computing costs required by the dual exterior-source/asymptotic-matching boundary-condition strategy we use.

5.4 Computational cost of the FC BVP solver

The computational cost of the proposed FC-AD method for the solution of time-dependent problems depends linearly on the cost of the evaluations of the ODE solution and evaluation mappings A_N and B_N defined in (37) and (38), respectively. Thus, an efficient numerical implementation of these operators translate into corresponding efficiencies for the resulting overall FC-AD time-marching scheme.

The evaluation of the mapping A_N comprises two main components, namely 1) Fourier continuation (as described in Section 3.3) of the variable coefficients and the right-hand side, and 2) Numerical solution of the resulting linear system (22). In view of Remark 4, point 1) requires $\mathcal{O}(N \log N)$ operations. With regards to point 2), on the other hand, we note that every iteration of the GMRES method involves one evaluation of the discrete ODE operator B_N and one evaluation of the preconditioner. Inspection of Algorithm 2 and the Finite Difference preconditioning algorithm presented in Section 5.1 therefore shows that the overall cost of the of the ODE solver is $2\text{FFT}(N + N_d - 1) + N_{\text{iter}}\text{FFT}(N_{\text{over}}(N + N_d - 1)) + 4N_{\text{iter}}\text{FFT}(2(N + N_d - 1)) + \mathcal{O}(N_{\text{over}}(N + N_d - 1))$, where $\text{FFT}(M)$ denotes the number of operations required to evaluate an FFT of size M . In brief, the ODE solver runs in $N_{\text{iter}}\mathcal{O}(N \log N)$ operations.

6 Full FC-based PDE solver

Based on algorithms introduced above in this text, this section introduces FC-based alternating-direction solvers for diffusion and wave propagation PDEs with variable coefficients.

6.1 Overall spatial discretization

Let Ω be a two-dimensional spatial domain with a piecewise smooth boundary which, without loss of generality, we assume is contained in the rectangle $R = [0, L^H] \times [0, L^V]$. In order to produce a spatial discretization of Ω , the rectangle R itself is discretized by means of a uniform Cartesian grid Ω_h given by

$$\Omega_h = \Omega \cap \{(x_i, y_j) = ((i-1)h^H, (j-1)h^V), i = 1, \dots, M^H, j = 1, \dots, M^V\},$$

where, for some integers $M^H, M^V > 1$ we have set $h^H = L^H/(M^H - 1)$ and $h^V = L^V/(M^V - 1)$. For the sake of simplicity we assume that each grid line only crosses the boundary $\partial\Omega$ twice, and we denote by

$$\begin{aligned} \{a_j^H, b_j^H\} &= \{x : (x, y_j) \in \partial\Omega\} & (j = 1, \dots, M^V), \quad \text{and} \\ \{a_i^V, b_i^V\} &= \{y : (x_i, y) \in \partial\Omega\} & (i = 1, \dots, M^H), \end{aligned}$$

the points of intersection of $\partial\Omega$ with horizontal and vertical Cartesian lines (shown as green circles and yellow squares in Figure 7), respectively. (Generalization to cases for which more than two intersections occur for some Cartesian lines is straightforward.) The horizontal and vertical discretization lines within Ω and the corresponding sets of indexes, in turn, are given by

$$\begin{aligned} P_j^H &= \{(x, y_j) \in \Omega_h\} & ; & \quad I_j^H = \{i \in \mathbb{N} : (x_i, y_j) \in \Omega_h\} & (j = 1, \dots, M^V), \quad \text{and} \\ P_i^V &= \{(x_i, y) \in \Omega_h\} & ; & \quad I_i^V = \{j \in \mathbb{N} : (x_i, y_j) \in \Omega_h\} & (i = 1, \dots, M^H). \end{aligned}$$

Each alternating-direction half-step in the time-marching algorithm presented in Section 2.2 consists of a horizontal sweep (steps (D2) and (W2)) and a vertical sweep (steps (D4) and (W3)). For each horizontal (resp. vertical) sweep, the solver requires solution of a Dirichlet two-point BVP along each vertical (resp. horizontal) line P_i^V (resp. P_j^H). The solution of each Dirichlet two-point BVP, in turn, involves application of the operators (38) (evaluation of the ODE right-hand side) and (37) (BVP solution) (see also Section 4.5 and Remarks 12 and 13). Sections 6.2 and 6.3 provide a detailed description of the discrete versions of the horizontal and vertical sweeps, for the problems of diffusion and wave propagation, in terms of the operators A_N and B_N .

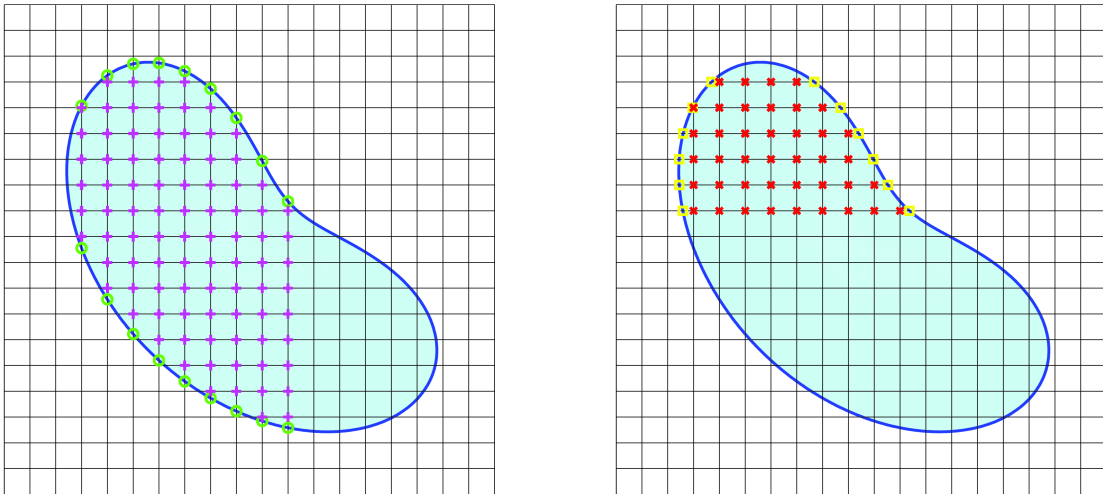


Figure 7: Sweeping procedure used for evaluation of the grid values $\mathbf{u}_{ij}^{n+\frac{1}{2}}$ on vertical lines (magenta “+” crosses on the left) and the grid values \mathbf{u}_{ij}^{n+1} on horizontal lines (red “x” crosses on the right).

Remark 17. Throughout Section 6, a bold-face symbol such as $\boldsymbol{\phi}_{i,j}^n$ denotes the value of a grid function at a point $(x_i, y_j) \in \Omega_h$ at time t_n , and for each fixed i (resp. for each fixed j), we define the vector $\boldsymbol{\phi}_{i,\cdot}^n = (\phi_{i,j}^n)_{j \in I_i^V}$ (resp. $\boldsymbol{\phi}_{\cdot,j}^n = (\phi_{i,j}^n)_{i \in I_j^H}$). In addition, in the case of the diffusion and wave problems, with reference to (13) and (14) respectively we set

$$\boldsymbol{p}_{i,j}^H = \mathcal{P}^H(x_i, y_j), \quad \boldsymbol{p}_{i,j}^V = \mathcal{P}^V(x_i, y_j), \quad \boldsymbol{q}_{i,j} = \mathcal{Q}(x_i, y_j), \quad \text{and} \quad \boldsymbol{f}_{i,j}^n = \mathcal{F}(x_i, y_j, t_n).$$

Since originally only the grid values of α and β are known, the grid values of $\partial_x \beta$ and $\partial_y \beta$ needed to evaluate equation \mathcal{P}^H and \mathcal{P}^V in (13) and (14) are approximated at each line by the derivative of its scaled FC(Gram) continuation.

6.2 FC-AD diffusion solver

Taking into account Sections 2.2.1, 4.5 and 6.1 (and, in particular, Remark 17 concerning bold-face symbols denoting grid functions and vectors of grid function values), our full FC-based ADI discrete procedure for the diffusion problem consists of the following four steps:

(D1)_N Initialize $\boldsymbol{u}_{i,j}^0$ and $\boldsymbol{w}_{i,j}^0$ on the vertical lines P_i^V ($j \in I_i^V$, $i = 1, \dots, M^H$),

$$\begin{cases} \boldsymbol{u}_{i,j}^0 = u_0(x_i, y_j), \\ \boldsymbol{w}_{i,\cdot}^0 = B_N(-\boldsymbol{p}_{i,\cdot}^V, -\boldsymbol{q}_{i,\cdot})\boldsymbol{u}_{i,\cdot}^0, \end{cases}$$

and, for $n = 0, 1, \dots, n_{\max}$,

(D2)_N compute $\boldsymbol{u}_{i,j}^{n+\frac{1}{2}}$ on the vertical lines P_i^V ($j \in I_i^V$, $i = 1, \dots, M^H$),

$$\boldsymbol{u}_{i,\cdot}^{n+\frac{1}{2}} = A_N(\boldsymbol{p}_{i,\cdot}^V, \boldsymbol{q}_{i,\cdot}) \begin{pmatrix} \boldsymbol{w}_{i,\cdot}^n + \boldsymbol{f}_{i,\cdot}^{n+\frac{1}{4}} \\ g^{n+\frac{1}{2}}(a_i^V) \\ g^{n+\frac{1}{2}}(b_i^V) \end{pmatrix},$$

(D3)_N compute $\boldsymbol{w}_{i,j}^{n+\frac{1}{2}}$ on the horizontal lines P_j^H ($i \in I_j^H$, $j = 1, \dots, M^V$),

$$\boldsymbol{w}_{i,j}^{n+\frac{1}{2}} = 2\boldsymbol{u}_{i,j}^{n+\frac{1}{2}} - \boldsymbol{w}_{i,j}^n - \boldsymbol{f}_{i,j}^{n+\frac{1}{4}},$$

(D4)_N compute $\boldsymbol{u}_{i,j}^{n+1}$ on the horizontal lines P_j^H ($i \in I_j^H$, $j = 1, \dots, M^V$),

$$\boldsymbol{u}_{\cdot,j}^{n+1} = A_N(\boldsymbol{p}_{\cdot,j}^H, \boldsymbol{q}_{\cdot,j}) \begin{pmatrix} \boldsymbol{w}_{\cdot,j}^{n+\frac{1}{2}} + \boldsymbol{f}_{\cdot,j}^{n+\frac{3}{4}} \\ g^{n+1}(a_j^H) \\ g^{n+1}(b_j^H) \end{pmatrix}.$$

Note that the initialization step (D1)_N only requires no more than M^H evaluations of the mapping B_N , and every time step (D2)_N-(D4)_N involves no more than $M^H + M^V$ evaluations of the mapping A_N —in other words, each time step requires no more than $M^H + M^V$ solutions of one-dimensional two-point BVP with Dirichlet boundary conditions. The computational cost of these ODE solvers is analysed in Section 5.4. The cost of the overall diffusion PDE solver (which, in brief, runs at FFT speeds), is illustrated in Tables 1 and 2.

6.3 FC-AD wave propagation solver

Taking into account Sections 2.2.2, 4.5 and 6.1 as well as Remark 17, our full FC-based ADI discrete procedure for the wave propagation problem consists of the following three steps:

(W1)_N Initialize $\mathbf{u}_{i,j}^0$ and $\mathbf{u}_{i,j}^1$ on the vertical lines P_i^V ($j \in I_i^V$, $i = 1, \dots, M^H$),

$$\begin{cases} \mathbf{u}_{i,j}^0 = u_0(x_i, y_j), \\ \mathbf{u}_{i,j}^1 = u_0(x_i, y_j) + \Delta t u_1(x_i, y_j), \end{cases}$$

and, for $n = 0, 1, \dots, n_{\max}$,

(W2)_N compute $\mathbf{w}_{i,j}^{n+\frac{1}{2}}$ on the vertical lines P_i^V ($j \in I_i^V$, $i = 1, \dots, M^H$),

$$\mathbf{w}_{i,\cdot}^{n+\frac{1}{2}} = A_N(\mathbf{p}_{i,\cdot}^V, \mathbf{q}_{i,\cdot}) \begin{pmatrix} 2\mathbf{u}_{i,\cdot}^n - \mathbf{u}_{i,\cdot}^{n-1} + \mathbf{f}_{i,\cdot}^{n+\frac{1}{2}} \\ g^{n+1}(a_i^V) \\ g^{n+1}(b_i^V) \end{pmatrix},$$

(W3)_N compute $\mathbf{u}_{i,j}^{n+1}$ on the horizontal lines P_j^H ($i \in I_j^H$, $j = 1, \dots, M^V$),

$$\mathbf{u}_{\cdot,j}^{n+1} = A_N(\mathbf{p}_{\cdot,j}^H, \mathbf{q}_{\cdot,j}) \begin{pmatrix} \mathbf{w}_{\cdot,j}^{n+\frac{1}{2}} \\ g^{n+1}(a_j^H) \\ g^{n+1}(b_j^H) \end{pmatrix}.$$

Every time step (W2)_N-(W3)_N in the present wave equation algorithm requires at most $M^H + M^V$ evaluations of the mapping A_N and thus, in view of Section 5.4, the overall FC-based wave propagation solver runs at FFT speeds.

7 Numerical results

This section presents a variety of numerical results demonstrating the accuracy, unconditional stability, reduced computational cost and spatial dispersionlessness of the variable-coefficient FC-AD algorithms introduced in this paper. Implementation details and hardware setup used include the following:

1. All numerical simulations presented in this section have resulted from Fortran implementations of the FC-AD algorithms introduced in previous sections, running on a single processor Intel® Xeon® (model X5570) at 2.93 GHz with 8MB cache size.
2. In accordance with Sections 5.1 and 5.2, the numerical simulations presented in this section use the FC(Gram) parameters $N_\Delta = 10$ and $N_d = \lfloor 26(1 + (N - 21)/100) \rfloor$, and, following [6, 12], the values $m = 5$ and 4 for PDE solver of the diffusion and the wave model, respectively. (In agreement with those references we have found that these values of m ensure unconditional stability of the FC-AD algorithm.) In all cases the oversampling parameter used for the GMRES preconditioner (Section 5.1) is set to $N_{\text{over}} = 4$, and boundary conditions are enforced by means of the hybrid exterior-source/asymptotic-matching algorithm (see Remark 16).
3. At each time step t_n the solution is stored as a matrix $(\mathbf{u}_{i,j}^{n+1})$ of size $M^H \times M^V$ containing the approximate solution values at $(x_i, y_j) \in \Omega_h$ and zeroes for $(x_i, y_j) \notin \Omega_h$.

4. Since the forward operator B_N and the BVP solver A_N use periodic extensions of the original problem, the Fortran implementation of both procedures for each horizontal and vertical grid lines use work vectors with more entries than the number of points supported on lines P_i^H and P_j^V of Ω_h . For each half-time step $((D2)_N, (W2)_N, (D4)_N$ and $(W3)_N$) and each associated horizontal and vertical line, only the output quantities for indices in the corresponding sets I_i^H and I_j^V (that is, for the corresponding points in the computational domain Ω_h) are stored in the matrices $(\mathbf{u}_{i,j}^k)$.
5. All needed FFTs are performed using the FFTW library [10]. In addition, FFTW have been used to transpose *in-place* the matrices $(\mathbf{u}_{i,j}^k)$ to preserve the contiguous memory access (column-wise in the Fortran implementation) prior to the needed transfers of horizontal or vertical lines to the BVP solver work vector mentioned in point 4.

7.1 Boundary conditions: exterior-source and asymptotic-matching procedures

As indicated in Remark 16, our variable-coefficient FC-AD algorithm automatically selects the mechanism—either exterior-sources or asymptotic-matching—for enforcement of boundary conditions. In view of the discussion in Section 5.3.2, it is clear that the hybrid exterior-source/asymptotic-matching procedure can be used to produce accurate solutions for arbitrary time-steps in computing times per time-step that remain bounded as $\Delta t \rightarrow 0$. The present section, in turn, presents numerical results that demonstrate quantitatively the impact of the hybrid approach, in terms of computing time and accuracy, on the solution of full PDE problems.

Table 1, which presents computing times required by the FC-AD diffusion solver for 1000×1000 two-dimensional grid and for two different values of the time step Δt (in the particular case of the first diffusion problem considered in Section 7.2, see also Figure 9), provides some insight into the computing costs required by the hybrid approach in each of the two possible $(h, \Delta t)$ regimes. The “Setup” columns in this and subsequent tables display the overall time used in precomputations—including each one of the following operations *for each horizontal and vertical line* P_j^H and P_i^V : 1) Precomputation of FFTW plans [10] in real-valued arithmetic; 2) Evaluation of the scaled FC(Gram) continuations for the variable coefficients and the right-hand side; 3) LU factorization of the preconditioning finite-difference matrix (see Section 5.1); 4) Evaluation of the auxiliary solutions for treatment of boundary conditions (see Section 4.2 and 4.3); and 5) Initialization of the initial time step. As can be gleaned from Table 1, for the diffusion problem under consideration, the FC-AD asymptotic-matching procedure leads to somewhat smaller overall setup times but comparable times per time-step as the corresponding exterior-source method; similar remarks apply to our FC-AD implementation of the variable-coefficient wave propagation problem. Thus, the asymptotic-matching method resolves the boundary layers that arise for small values of Δt (which cannot be accurately discretized by the exterior-source method, unless unduly fine grids are used) at a cost comparable to that which would be required by the exterior-source method in absence of boundary layers. The exterior-source method, in turn, can adequately treat cases in which no significant boundary layers exist—for which the asymptotic-matching method would be inaccurate; cf. Figure 6 and 8.

To demonstrate, in a simple context, the hybrid exterior-source/asymptotic-matching procedure for enforcement of boundary conditions (Sections 2.2.2, 4.2 and 4.3), here we apply the hybrid algorithm in conjunction with a one-dimensional FC-based time marching scheme to a one-dimensional wave-propagation problem: the one-dimensional wave equation (2) with variable coefficients $\alpha(x) = 1 + 4x^2$ and $\beta(x) = 2 - x + 8x^2$ in the unit interval, and with Dirichlet boundary conditions such that the exact solution is given by the oscillatory function $u(x, t) = \sin(100x - 2\pi t)$. Figure 8 displays the resulting maximum relative solution error as a function of the time step Δt and the grid size h . The left error map in this figure presents the relative errors that result as the exterior-source procedure is used for all values of $(h, \Delta t)$. As expected, in presence of boundary layers (that arise for small values of Δt), the overall numerical approximation provided by the exterior-source algorithm is completely inaccurate. The right-hand error map in Figure 8, on the other hand, displays the errors that result from the hybrid boundary-conditions algorithm. The dashed line in the

Boundary condition enforcement	N_{over}	$tol_{\text{GMRES}} = 10^{-15}$		$tol_{\text{GMRES}} = 10^{-10}$		$tol_{\text{GMRES}} = 10^{-6}$	
		Setup	Time step	Setup	Time step	Setup	Time step
$h = 10^{-3}$,	1	13.493	6.169	5.617	2.265	4.264	1.387
$\Delta t = 5 \times 10^{-3}$,	2	8.119	3.576	5.693	2.129	4.943	1.480
(exterior-sources	4	8.628	3.788	6.343	2.332	5.408	1.816
regime)	8	11.700	5.189	8.816	3.273	7.532	2.548
$h = 10^{-3}$,	1	3.039	6.301	3.005	3.339	3.041	1.478
$\Delta t = 10^{-6}$,	2	3.182	5.976	3.142	2.474	3.180	1.626
(asymptotic-matching	4	4.175	6.326	4.147	2.785	4.183	1.848
regime)	8	5.499	8.275	5.456	3.630	5.496	2.585

Table 1: CPU time (in seconds) required by our implementation of the FC-AD method, for the variable-coefficient diffusion problem mentioned in Section 7.1, in two different regimes of the boundary condition enforcement algorithm.

right-hand map separates regimes in the hybrid approach: above this line the exterior-source algorithm was utilized, below this line the asymptotic-matching method was used. Close consideration of the right-hand error map shows that the hybrid boundary-conditions algorithm leads to an overall convergent PDE solver. In the right-most region of the right-hand error map (larger values of h), the error, which results mostly from the coarseness of the spatial FC discretization, is essentially independent of Δt . In the left-most portion of the map, where finer spatial grids are used, the spatial discrete errors are negligible in comparison with the time discretization errors, and hence the resulting errors depends only on the time-step value Δt .

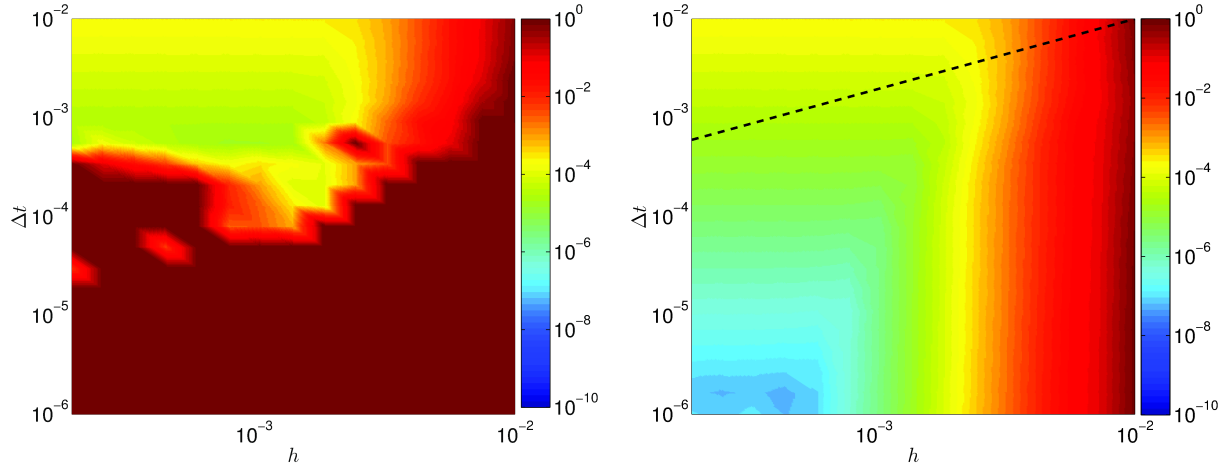


Figure 8: Maximum relative error in the FC-AD approximate solution of a one dimensional wave equation of the form (2) using solely the exterior-source procedure (left) and using the hybrid boundary-conditions algorithm (right).

7.2 Diffusion problem: performance, convergence and stability

A variety of numerical examples presented in this section demonstrate the character of the FC-AD scheme for diffusion problems with variable coefficients. For the first set of tests of this section we consider a problem of the form (1) with variable coefficients given by $\alpha(x, y) = x + y + 1$ and $\beta(x, y) = 2x + 0.5y + 1$ in the domain

bounded by the curve $(x/9)^6 + (y/5)^6 = (1/20)^6$. The right-hand side and the boundary conditions have been selected in such a way that the function $u(x, y, t) = \sin(\pi(3x^2 + 2y^2 + 2t))$ is the exact solution of the problem. For our Δt convergence studies the FC-AD numerical solution is produced up to the final time $T = 0.1$. The

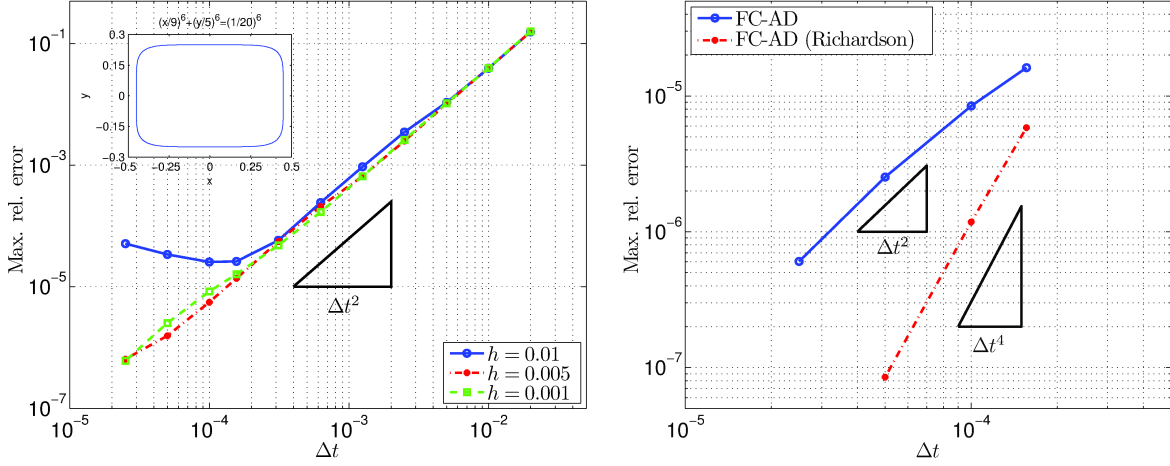


Figure 9: Left: Maximum relative errors in the FC-AD approximate solution of the first diffusion problem mentioned in Section 7.2, as a function of the time step Δt , for three different spatial grids. Right: Comparison of these solution errors with those obtained, for the same problem, via an application of the Richardson extrapolation method leading to two orders of improvement in the temporal convergence rate.

spatial discretizations are given by uniform grids of size h within the square $[-0.5, 0.5] \times [-0.3, 0.3]$, whereas the preconditioner uses the oversampling ratio $N_{\text{over}} = 4$ and the GMRES tolerance $\text{tol}_{\text{GMRES}} = 10^{-10}$.

Figure 9 displays maximum relative errors throughout Ω_h (evaluated through comparison with the exact solution) produced by the second-order in time FC-AD algorithm described in Section 6.2 for $h = 0.01$, 0.005 , and 0.001 (left plot in Figure 9), as well as corresponding results obtained by means an additional application of the Richardson extrapolation procedure (right plot in Figure 9, see [6] and references therein for details on the the application of the Richardson extrapolation method in the time domain). Since the spatial discretization errors for different grids are negligible with respect to the time discretization (at least for the the coarser time steps), the relative error exhibits the expected second-order convergence that results from the ADI time-marching scheme described in Section 2.2. Additionally, the convergence displayed in this figure demonstrates that the underlying solver is not subject to the ordinary CFL constraint $\Delta t \sim h^2$ required by explicit solvers: for the case $h = 0.001$ the largest Δt values used here are in fact four orders of magnitude larger than would be allowed by the quadratic CFL constraint. (In fact, our experiments suggest that Δt can be increased arbitrarily without leading to instability: values as large as $\Delta t = 100$ and $\Delta t = 1000$, etc, lead to stable, albeit inaccurate solutions.)

To demonstrate the performance of the FC-AD method for general geometries we consider the curved, non-convex *heating circuit* structure depicted in Figure 10. The variable heat-equation coefficients used correspond to (variable) thermal constants of silicon; the geometry, in turn, represents a 80×120 rectangular plate containing a curved heating circuit. The thermal conductivity varies from 5 to 50 (the typical range for silicon) as shown in the left plot of Figure 10. The initial temperature has been fixed to $u_0 = 55$. The temperature profile on the exterior boundaries of the plate is fixed also to 55, whereas that the inner boundaries of the heating circuit are driven by the function $g(x, y, t) = 55 + 30 \sin(20\pi t)$.

The right plot in Figure 10 presents the temperature field at time $T = 0.1$ —that is, one period of the boundary data function g —evaluated by means of the FC-AD algorithm described in Section 6.2 (without Richardson extrapolation, for simplicity) on a spatial grid containing 800×1300 discretization points. The

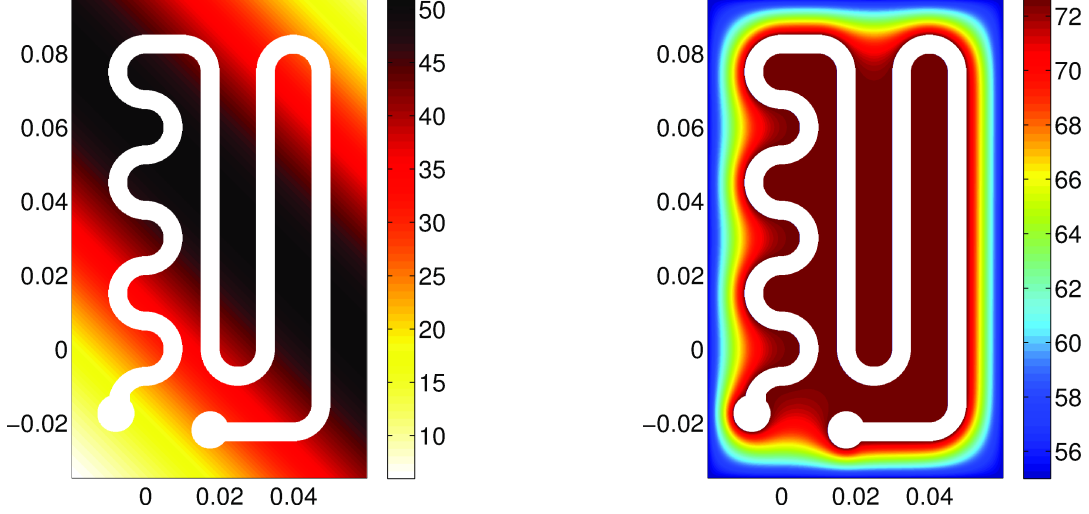


Figure 10: Variable thermal conductivity (left) and temperature field produced by the FC-AD method at time $T = 0.1$ (right) for the second diffusion problem mentioned in Section 7.2.

corresponding maximum errors were evaluated through comparison with a reference numerical solution produced using $\Delta t = 2.5 \times 10^{-5}$ on a 1600×2600 spatial grid: it was found that the 800×1300 solutions with $\Delta t = 10^{-4}$ and 5×10^{-5} and $tol_{\text{GMRES}} = 10^{-10}$ contain maximum errors of 0.054% and 0.013%, respectively—demonstrating, in particular, second order convergence in time. Analogous relative errors at somewhat faster computing times result from use of the tolerance value $tol_{\text{GMRES}} = 10^{-6}$.

Δt	N_{over}	$tol_{\text{GMRES}} = 10^{-10}$			$tol_{\text{GMRES}} = 10^{-6}$		
		Setup	Time step	Total	Setup	Time step	Total
10^{-4}	4	33.989	2.343	268.289	25.798	2.107	236.498
5×10^{-5}	4	33.696	2.406	274.296	23.944	1.883	212.244
2.5×10^{-5}	4	34.095	2.371	271.195	23.271	1.693	192.571

Table 2: CPU times (in seconds) required to evolve the heating-circuit FC-AD solver for a total 100 time steps on a 800×1300 grid for various values of the time step Δt .

The CPU times required to obtain the various heating-circuit solutions mentioned above are presented in Table 2 for two different values of the GMRES residual tolerance tol_{GMRES} (both of which yield similar errors, as indicated previously in this section, for the values the parameters under consideration presently).

7.3 Wave propagation problem: performance, convergence and stability

The following two subsections demonstrate the properties of the FC-AD solver for the wave equation with variable coefficients. Section 7.3.1 highlights the highly significant performance gains that arise from the low spatial dispersion inherent in the FC methodology, including comparisons with finite-difference methods. Section 7.3.2, in turn, presents an application of the FC solver to a non-trivial geometry, and it demonstrates the convergence and stability of the approach.

7.3.1 Spatial dispersionlessness

We demonstrate the spatial dispersionlessness of our FC wave-equation solvers by means of the one-dimensional problem defined by the equation $\alpha(x)\partial_{tt}u - \partial_x(\beta(x)\partial_x u) = 0$ with variable coefficients $\alpha(x) = 1 + x/2$ and $\beta(x) = 2/(2 + x)$ in the spatial interval $(a, b) = (0, 1)$, with final time $T = 1$, and with Dirichlet boundary conditions such that the exact solution is given by $u(x) = \sin(2\pi f(x^2/4 + x + t))$. The FC-based time marching scheme for the present one-dimensional case is analogous to that presented in Section 2.2.2 for two dimensions but, of course, in here use of alternating directions is neither possible nor necessary. In order to focus attention on the spatial dispersion properties of the algorithm, our first example in this section uses a fixed time discretization which gives rise to errors smaller than all the corresponding spatial errors: $\Delta t = 8 \times 10^{-7}$ used in conjunction with second order Richardson extrapolation. The spatial grids used, in turn, are frequency dependent: they are taken to hold a prescribed number of points per wavelength (PPW). Since the coefficients and thus the wavelength vary within the physical domain, the PPW quantity is defined as the number of discretization points used within the *shortest* wavelength—which, in our case, can be defined as the minimum value of the “wavelength function” $\min\{1/(f(x/2 + 1)) : x \in [0, 1]\} = 2/(3f)$.

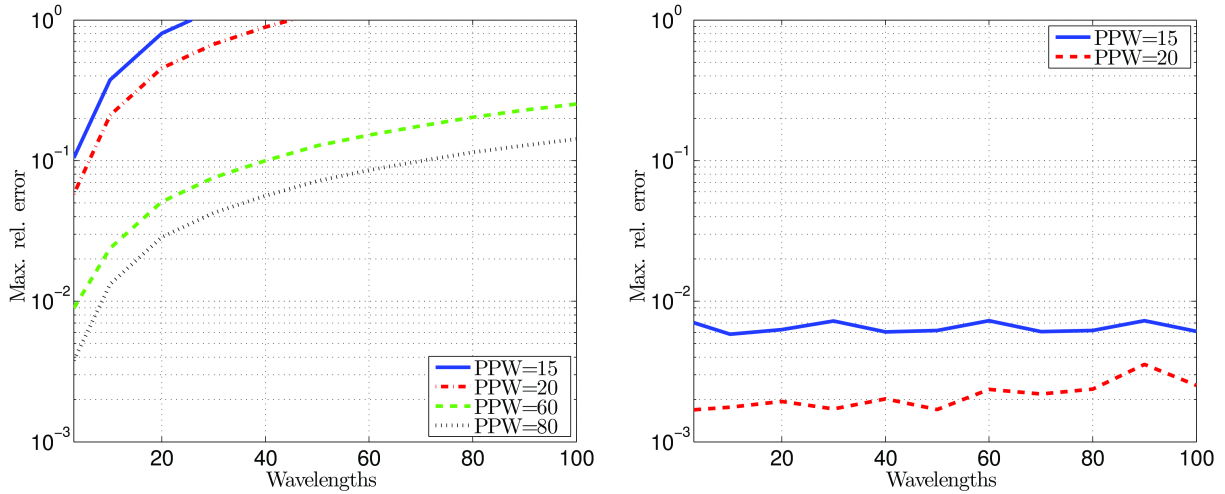


Figure 11: Maximum relative errors in numerical solutions of the one-dimensional wave equation in the time interval $[0, 1]$ using a fixed number of Points-Per-Wavelength (PPW). Left: one-dimensional finite difference solver. Right: one-dimensional FC-AD solver.

Figure 11 displays maximum errors relative to the maximum solution values for numerical solutions produced by two spatial differentiation methods, namely second-order finite differences (left portion of Figure 11) and Fourier continuation (right portion of Figure 11). The time discretization in both cases is the one described above: second-order Richardson extrapolation with $\Delta t = 8 \times 10^{-7}$. It can be seen from these figures that, even at the lowest frequencies (lowest number of wavelengths in the domain $(0, 1)$), the FC solutions for 15 and 20 PPW are significantly more accurate than the corresponding finite-difference solutions, and, for such low frequencies, only the finite-difference solutions using 60 and 80 PPW approximately match the error in the 15 and 20 PPW FC solutions. As the frequency increases the numerical error in the FC solutions remains essentially constant, that is, the FC solver is essentially spatially dispersionless. The finite difference scheme, on the other hand, is not: numerical errors increase significantly with frequency, and an 80 PPW mesh can only produce 1% accurate solutions for problems containing no more than five PPW.

7.3.2 Wave propagation problem: complex structures

This section demonstrates the properties of the overall FC-AD scheme for problems of wave motion with variable coefficients. In our first example we consider the wave propagation problem (2) with variable coefficients given by $\alpha(x, y) = 1 + x + y$, $\beta(x, y) = 2.0x + 0.5y + 1$ in the domain bounded by the curve $(x/9)^6 + (y/5)^6 = (1/20)^6$. The PDE right-hand side and boundary conditions are selected in such a way that the function $u(x, y, t) = \sin(\pi(x + 2y - t))$ is the exact solution of the problem. Uniform meshes of

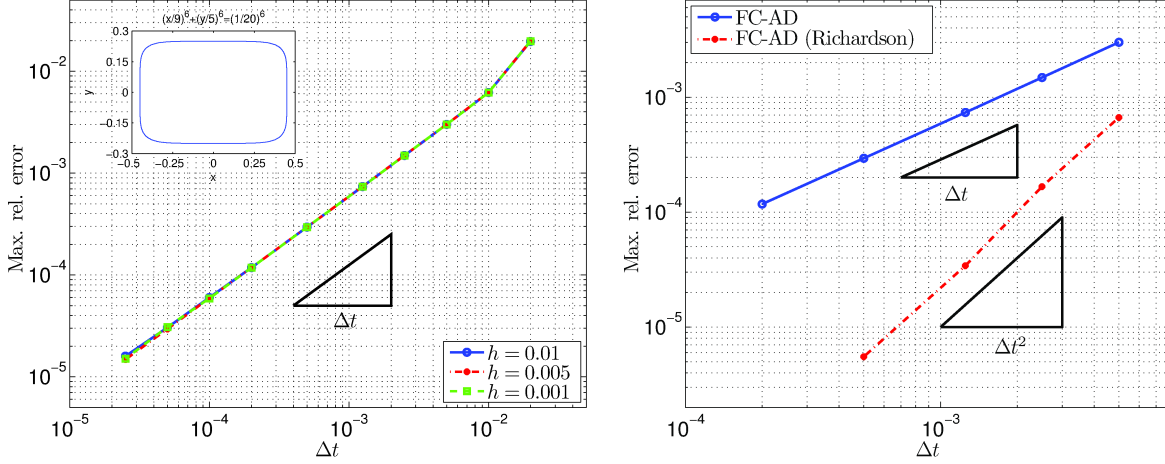


Figure 12: Left: Maximum relative errors in the FC-AD approximate solution of the first wave propagation problem mentioned in Section 7.3.2, as a function of the time step Δt , for three different spatial grids. Right: Comparison of these solution errors with those obtained, for the same problem, via an application of the Richardson extrapolation method leading to one order of improvement in the temporal convergence rate.

grid-size h are used to discretize the square $[-0.5, 0.5] \times [-0.3, 0.3]$ which contains the PDE domain. For our Δt convergence studies, errors in the FC-AD numerical solution are evaluated at the final time $T = 0.1$. The GMRES tolerance is set to $tol_{\text{GMRES}} = 10^{-10}$, and the preconditioner uses the oversampling ratio $N_{\text{over}} = 4$.

Figure 12 presents maximum values of the solution error throughout Ω_h (evaluated through comparison with the exact solution) relative to the maximum value of the solution, that results from use of the FC-AD algorithm described in Section 2.2.2 for $h = 0.01, 0.005$, and 0.001 (left plot), as well as corresponding results obtained by means an additional application of the Richardson extrapolation procedure (right plot). Since the spatial discretization errors for different grids are negligible with respect to the time discretization, the relative error exhibits the expected first and second-order convergence that results from the FC-AD time-marching scheme (described in Section 2.2) and the application of the Richardson extrapolation procedure, respectively. Once again, the unconditional stability of the FC-AD time-marching scheme allows us to consider a wide range of time steps, without CFL-type restrictions. Notice that, as a result of the hybrid exterior-source/asymptotic-matching procedure for enforcement of boundary conditions, small values of the time-step Δt (and the associated boundary layers) do not give rise to accuracy losses even when the coarse mesh-size $h = 0.01$ is used.

To illustrate the applicability of the FC-AD wave solver for PDEs with general spatially variable coefficients, we consider the *waveguide* depicted in the left portion of Figure 13, which consists of a pair of curved channels within a rectangular plate of dimensions 1×2 with chamfered corners. With reference to equation (2), the material is characterized by variable coefficients α and β where α varies between 1 and 20 as depicted in the left portion of Figure 13, and where $\beta = 1$. The structure is excited by a spatially Gaussian harmonic pulsed source of frequency $f = 3\pi$ supported in a disk of radius 0.05 centered at point $(0.5, 1)$. The solution at the final time $T = 4$, which is displayed on the right portion of Figure 13, was

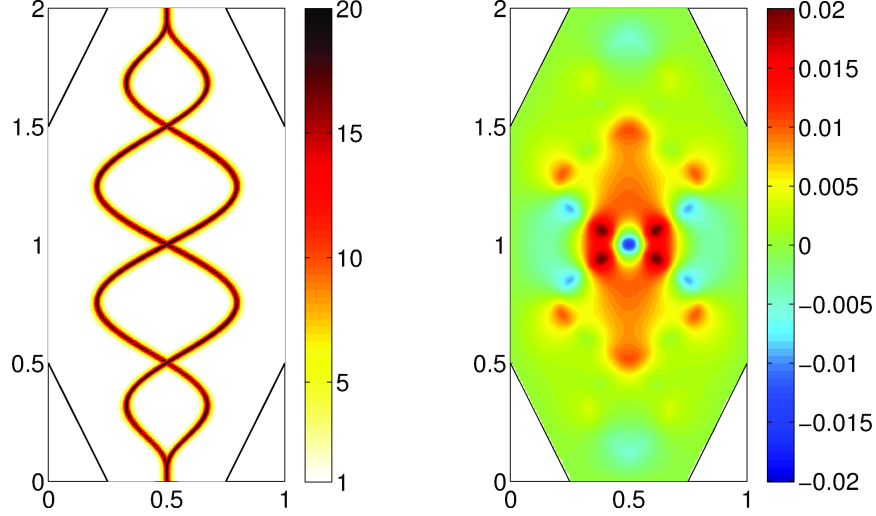


Figure 13: Left: Values of the variable coefficient $\alpha = \alpha(x, y)$. Right: FC-AD approximate solution at time $T = 4$.

obtained using $\Delta t = 10^{-2}$ on a 200×400 spatial grid and $tol_{\text{GMRES}} = 10^{-6}$ in a total computational time of 194.147 seconds (requiring 18.147 for the setup and 0.440 seconds per time-step). Using, for reference, a solution computed via a 400×800 spatial grid and $\Delta t = 2.5 \times 10^{-3}$, it was found that the solution above contains an error of 0.002%.

Conclusions

We have introduced Fourier-based alternating direction time-marching schemes for the numerical solution of linear PDEs *with variable coefficients*. Following [6, 12], our use of the Fourier continuation method in combination with the ADI time-marching scheme and the Fast Fourier Transform, gives rise to a highly desirable combination of properties, namely, unconditional stability and high-order accuracy and spatial dispersionlessness at FFT speeds in the general context of non-periodic functions and for general domains. A variety of numerical results demonstrate the properties of the resulting solvers for problems of diffusion as well as wave propagation and scattering in media with spatially varying characteristics.

A An auxiliary lemma

Lemma 1. Let \tilde{q}^ℓ , g_a and g_b be smooth functions defined in the interval $[b, c]$, and let \tilde{q}^ℓ be strictly positive in that interval. If g_a and g_b satisfy the conditions (25), then the overdetermined ODE problem

$$v - \tilde{p} \frac{dv}{dx} - \tilde{q}^\ell \frac{d^2v}{dx^2} = g_a + \mu g_b \quad \text{in } (b, c), \quad (\text{A.1})$$

$$v(b) = \frac{dv}{dx}(b) = 0, \quad v(c) = \frac{dv}{dx}(c) = 0, \quad (\text{A.2})$$

is not solvable: equations (A.1)–(A.2) do not admit solutions v for any real value of the constant μ .

Proof. Assume a solution v of the problem (A.1)–(A.2) exists. Denoting by $G(x, \xi)$ the Green function of

the problem,

$$\begin{aligned} G(x, \xi) - \tilde{p} \frac{\partial}{\partial x} G(x, \xi) - \tilde{q}^\ell \frac{\partial^2}{\partial x^2} G(x, \xi) &= \delta_{(x=\xi)} \quad \text{in } (b, c), \\ G(b, \xi) &= G(c, \xi) = 0, \end{aligned}$$

and letting

$$h = g_a + \mu g_b, \tag{A.3}$$

the solution v can be expressed in the form

$$v(x) = \int_b^c G(x, \xi) h(\xi) d\xi.$$

Taking into account the Neumann boundary conditions (A.2) we then obtain

$$\frac{dv}{dx}(b) = \int_b^c \frac{\partial G}{\partial x}(b, \xi) h(\xi) d\xi = 0, \tag{A.4}$$

$$\frac{dv}{dx}(c) = \int_b^c \frac{\partial G}{\partial x}(c, \xi) h(\xi) d\xi = 0. \tag{A.5}$$

Now, as is known (see e.g. in [17, Ch. V.28]), the function $\partial G / \partial \xi$ satisfies the ODE problems

$$\begin{aligned} \frac{\partial G}{\partial \xi}(x, b) - \tilde{p} \frac{\partial}{\partial x} \left(\frac{\partial G}{\partial \xi}(x, b) \right) - \tilde{q}^\ell \frac{\partial^2}{\partial x^2} \left(\frac{\partial G}{\partial \xi}(x, b) \right) &= 0 \quad \text{in } (b, c), \\ \frac{\partial G}{\partial \xi}(b, b) &= \frac{1}{\tilde{q}^\ell(b)}, \quad \frac{\partial G}{\partial \xi}(c, b) = 0 \end{aligned}$$

and

$$\begin{aligned} \frac{\partial G}{\partial \xi}(x, c) - \tilde{p} \frac{\partial}{\partial x} \left(\frac{\partial G}{\partial \xi}(x, c) \right) - \tilde{q}^\ell \frac{\partial^2}{\partial x^2} \left(\frac{\partial G}{\partial \xi}(x, c) \right) &= 0 \quad \text{in } (b, c), \\ \frac{\partial G}{\partial \xi}(b, c) &= 0, \quad \frac{\partial G}{\partial \xi}(c, c) = -\frac{1}{\tilde{q}^\ell(c)}. \end{aligned}$$

In view of the identity $\frac{\partial G}{\partial x}(x, \xi) = \frac{\partial G}{\partial \xi}(\xi, x)$ (which follows from the symmetry $G(x, \xi) = G(\xi, x)$ of the Green function) it follows that the function

$$H(x) = \frac{\partial G}{\partial x}(b, x) - \frac{\partial G}{\partial x}(c, x)$$

satisfies the two-point boundary-value problem

$$\begin{aligned} H - \tilde{p} \frac{dH}{dx} - \tilde{q}^\ell \frac{d^2 H}{dx^2} &= 0 \quad \text{in } (b, c), \\ H(b) &= \frac{1}{\tilde{q}^\ell(b)}, \quad H(c) = \frac{1}{\tilde{q}^\ell(c)}. \end{aligned}$$

Applying the strong maximum principle [9] to this elliptic equation we obtain the estimate

$$H(x) = \frac{\partial G}{\partial x}(b, x) - \frac{\partial G}{\partial x}(c, x) \geq C > 0 \quad \text{for } x \in [b, c],$$

where C is the strictly positive constant $C = \min\{1/\tilde{q}^\ell(b), 1/\tilde{q}^\ell(c)\}$. From (25), (A.3), (A.4) and (A.5) we thus obtain

$$0 = \int_b^c \left(\frac{\partial G}{\partial x}(b, x) - \frac{\partial G}{\partial x}(c, x) \right) h(x) dx \geq C \int_b^c (g_a(x) + \mu g_b(x)) dx = C \int_b^c g_a(x) dx > 0,$$

which is a contradiction, and the lemma follows. \square

References

- [1] C. M. Bender and S. A. Orszag. *Advanced mathematical methods for scientists and engineers: Asymptotic methods and perturbation theory*, volume 1. Springer Verlag, 1978.
- [2] J. P. Boyd. *Chebyshev and Fourier spectral methods*. Dover Publications, 2001.
- [3] J. P. Boyd. A comparison of numerical algorithms for Fourier extension of the first, second, and third kinds. *J. Comput. Phys.*, 178(1):118–160, 2002.
- [4] O. P. Bruno and N. Albin. A spectral FC solver for the compressible Navier-Stokes equations in general domains I: Explicit time-stepping. *J. Comput. Phys.*, 230(16):6248–6270, 2011.
- [5] O. P. Bruno, Y. Han, and M. M. Pohlman. Accurate, high-order representation of complex three-dimensional surfaces via Fourier continuation analysis. *J. Comput. Phys.*, 227(2):1094–1125, 2007.
- [6] O. P. Bruno and M. Lyon. High-order unconditionally stable FC-AD solvers for general smooth domains I. Basic elements. *J. Comput. Phys.*, 229(6):2009–2033, 2010.
- [7] O.P. Bruno. Fast, high-order, high-frequency integral methods for computational acoustics and electromagnetics. In M. Ainsworth, P. Davies, D. Duncan, P. Martin, and B. Rynne, editors, *Topics in Computational Wave Propagation Direct and Inverse Problems Series*, volume 31 of *Lecture Notes in Computational Science and Engineering*, pages 43–82, 2003.
- [8] J. Douglas, Jr. and H. H. Rachford, Jr. On the numerical solution of heat conduction problems in two and three space variables. *T. Am. Math. Soc.*, 82(2):421–439, 1956.
- [9] L. C. Evans. *Partial Differential Equations*, volume 12 of *Graduate Studies in Mathematics*. American Mathematical Society, 1998.
- [10] M. Frigo and S. G. Johnson. The design and implementation of FFTW 3. *Proc. IEEE Micr. Elect.*, 93(2):216–231, 2005.
- [11] J. Hesthaven, S. Gottlieb, and D. Gottlieb. *Spectral methods for time-dependent problems*. Cambridge University Press, 2007.
- [12] M. Lyon and O. P. Bruno. High-order unconditionally stable FC-AD solvers for general smooth domains II. Elliptic, parabolic and hyperbolic PDEs; theoretical considerations. *J. Comput. Phys.*, 229(9):3358–3381, 2010.
- [13] G. I. Marchuk. Finite Difference Methods: Splitting and Alternating Direction Methods. In P. G. Ciarlet and J. L. Lions, editors, *Handbook of numerical analysis*, volume 1, pages 197–462. North-Holland, 1990.
- [14] D. W. Peaceman and H. H. Rachford Jr. The numerical solution of parabolic and elliptic differential equations. *J. Soc. Ind. Appl. Math.*, 3(1):28–41, 1955.
- [15] W. Sun, W. Huang, and R. D. Russell. Finite difference preconditioning for solving orthogonal collocation equations for boundary value problems. *SIAM J. Numer. Anal.*, 33(6):2268–2285, 1996.
- [16] H. F. Walker. Implementation of the GMRES method using Householder transformations. *SIAM J. Sci. Stat. Comp.*, 9:152–163, 1988.
- [17] H. F. Weinberger. *A first course in partial differential equations with complex variables and transform methods*. Dover, 1995.